Computability Theory Notes*

Ashutosh Kumar

March 21, 2024

1 Recursive functions

Definition 1.1. *A finitary function on ω is an n-ary function $f : \omega^n \to \omega$ for some $1 \leq n < \omega$. The set of* ***primitive recursive functions****, denoted* PRec*, is defined to be the smallest set of finitary functions on ω satisfying the following.*

1. *(Identically Zero) Every $f : \omega^n \to \omega$ defined by $f \equiv 0$ is in* PRec.
2. *(Projections) For each $1 \leq k \leq n$, the function $f : \omega^n \to \omega$ defined by $f(x_1, \ldots, x_n) = x_k$ is in* PRec.
3. *(Successor function) $f : \omega \to \omega$ defined by $f(x) = x + 1$ is in* PRec.
4. *(Compositions) If $f : \omega^n \to \omega$ is in* PRec *and for each $1 \leq k \leq n$, $g_k : \omega^m \to \omega$ is in* PRec*, then $h : \omega^m \to \omega$ is in* PRec *where h is defined by*
$$h(x_1, \ldots, x_m) = f(g_1(x_1, \ldots, x_m), g_2(x_1, \ldots, x_m), \ldots, g_n(x_1, \ldots, x_m))$$
5. *(Recursion) If $g : \omega^{n+1} \to \omega$ and $h : \omega^{n-1} \to \omega$ are both in* PRec*, then $f : \omega^n \to \omega$ is in* PRec *where*
$$f(x_1, x_2, \ldots, x_n) = \begin{cases} h(x_2, \ldots, x_n) & \text{if } x_1 = 0 \\ g(f(x_1 - 1, x_2, \ldots, x_n), x_1 \ldots, x_n) & \text{if } x_1 \geq 1 \end{cases}$$

Definition 1.2. *An n-ary relation $R \subseteq \omega^n$ is primitive recursive iff its characteristic function is primitive recursive.*

Most elementary functions/relations that arise in arithmetic are primitive recursive. For example, addition, multiplication, exponentiation, factorial, Primes. The following is a useful tool in showing these facts (see Prop. 2.2 in Miller's notes).

Exercise 1.3. *Suppose $R(\vec{x}, y)$ is an $(n + 1)$-ary primitive recursive relation on ω and $f : \omega^n \to \omega$ is primitive recursive. Then $(\exists y \leq f(\vec{x}))(R(\vec{x}, y))$ is also primitive recursive.*

Observe that the set of primitive recursive functions is countable so most functions $f : \omega^n \to \omega$ are not primitive recursive. It is not difficult to convince oneself that *every primitive recursive function is computable in the sense that one can write a computer program that computes it.*

Is every computable function also primitive recursive? The answer is no. Let us see why. To every primitive recursive function f, one can associate a "certificate" C which shows how f was built from the basic functions (identically zero, projections and successor) using a finite number of applications of compositions and recursion. We can enumerate all of these certificates in a computable way as $C_1, C_2, C_3, \ldots$. Now define a function $f : \omega \to \omega$ as follows. If C_x is a certificate of a unary primitive recursive function $g : \omega \to \omega$, then $f(x) = g(x) + 1$. Otherwise, $f(x) = 0$. It should be intuitively clear that f is computable in the sense that one could write a computer program to compute it. We claim that f is not primitive recursive. Suppose it is. Then f has a certificate C. Since every certificate appears in the list $C_1, C_2, \ldots$, we can find an x such that $C = C_x$. Now by definition, $f(x) = f(x) + 1$: A contradiction. So f is not primitive recursive. A more natural example follows.

*These notes will be periodically revised. If you find an error, please let me know at krashu@iitk.ac.in

Exercise 1.4. *The Ackermann function $A : \omega^2 \to \omega$ is defined as follows: $A(0, n) = n + 1$, $A(m + 1, 0) = A(m, 1)$ and $A(m + 1, n + 1) = A(m, A(m + 1, n))$. Show that A is not primitive recursive.*

Definition 1.5. *A partial finitary function on ω is a function f such that $dom(f) \subseteq \omega^n$ and $range(f) \subseteq \omega$ for some $1 \le n < \omega$. The set of **general recursive functions**, denoted GRec, is defined to be the smallest set of partial finitary functions on ω that satisfies the following.*

1. *Every primitive recursive function is in GRec.*

2. *(Compositions) If f is an n-ary function in GRec and for each $1 \le k \le n$, g_k is an m-ary function in GRec, then h is in GRec where h is defined by*

$$h(x_1, \dots, x_m) = f(g_1(x_1, \dots, x_m), g_2(x_1, \dots, x_m), \dots, g_n(x_1, \dots, x_m))$$

3. *(Primitive recursion) If $g, h \in GRec$ where g is $(n + 1)$-ary and h is $(n - 1)$-ary, then f is in GRec where*

$$f(x_1, x_2, \dots, x_n) = \begin{cases} h(x_2, \dots, x_n) & \text{if } x_1 = 0 \\ g(f(x_1 - 1, x_2, \dots, x_n), x_1 \dots, x_n) & \text{if } x_1 \ge 1 \end{cases}$$

4. *(Unbounded search) If $g \in GRec$ is an $(n+1)$-ary, then $f \in \mathsf{GRec}$ where f is an n-ary partial function on ω defined by: $f(x_1, \dots, x_n) = z$ iff $g(z, x_1, \dots, x_n) = 0$ and for every $y < z$, $g(y, x_1, \dots, x_n)$ is defined and is nonzero.*

We saw that there is a computable function $f : \omega \to \omega$ that is not primitive recursive. The proof of this used diagonalization to produce a computable function which disagreed with every primitive recursive function on some input. Let us try to produce such a proof for the class of general recursive functions.

As before, we can associate to every general recursive function f, a certificate C which describes how f was built from the basic functions using a finite number of applications of compositions, primitive recursion and unbounded search. Let $C_1, C_2, \dots$ be a computable listing of all such certificates. As before, define a partial unary function f on ω as follows: If C_x is the certificate of a unary general recursive function g, then $f(x) = 1 + g(x)$. Clearly, f is a partial computable function. Let us assume that f is general recursive and try to get a contradiction. Fix x such that C_x is a certificate of f. Now **if** $x \in \mathrm{dom}(f)$, **then** $f(x) = 1 + f(x)$ which is impossible. So the only thing we can conclude here is that $x \notin \mathrm{dom}(f)$ which is not a contradiction. One could try to modify this argument by insisting that $C_1, C_2, \dots$ be a list of certificates of only total computable functions. But it is not clear at all if we can list them in a computable way. We will later show that we can't.

Exercise 1.6. *Convince yourself that the Ackermann function is general recursive.*

2 Turing machines

$X^{<\omega}$ is the set of all finite sequences of members of X. It is sometimes also denoted by $X^\star$ (Kleene star notation). For $\sigma \in X^\star$, we denote the length of σ by $|\sigma|$ or $\mathsf{length}(\sigma)$.

Definition 2.1. *An **alphabet** Σ is a set of symbols. A **string/word** over Σ is a member of $\Sigma^{<\omega}$. The **empty string** is denoted by $\langle\rangle$. The **concatenation of two strings** σ **and** τ, denoted $\sigma^\frown\tau$, is the string obtained by listing the entries of τ after σ.*

Example: Let $\Sigma = \{0, 1\}$. Then $\Sigma^{<\omega} = \{\langle\rangle, \langle 0\rangle, \langle 1\rangle, \langle 0, 0\rangle, \langle 0, 1\rangle, \dots\}$.

Definition 2.2. *Let Σ be an alphabet. We say that L is a **language over** Σ iff $L \subseteq \Sigma^{<\omega}$.*

Example: Let Σ be an alphabet. Define $Palindrome(\Sigma) = \{\sigma \in \Sigma^{<\omega} : \sigma = r(\sigma)\}$ where $r(\sigma)$ is the string obtained by listing the entries in σ in reverse order. For example, $r(\langle a, b, c\rangle) = \langle c, b, a\rangle$.

Definition 2.3. *A Turing machine T consists of the following.*

(1) A finite alphabet Σ with a blank symbol $B \in \Sigma$.

(2) A finite set S whose members of S are called states of T.

(3) A start state $q_s \in S$ and a halting state $q_h \in S$.

(4) A transition function τ that is a partial function from $\Sigma \times S$ to $\Sigma \times S \times \{-1, 1\}$.

Let $T = (\Sigma, S, B, q_s, q_h, \tau)$ be a TM. A **tape configuration** is a function $c : \mathbb{Z} \to \Sigma$ ($\mathbb{Z}$ is the set of all integers) such that $\{n \in \mathbb{Z} : c(n) \neq B\}$ is finite. Given an initial tape configuration $c_\star : \mathbb{Z} \to \Sigma$, the **run of T on $c_\star$** is a sequence $\langle (k_n, c_n, q_n) : n < \omega \rangle$ defined as follows.

(a) $(k_0, c_0, q_0) = (0, c_\star, q_s)$.

(b) If either $q_n = q_h$ or $(c_n(k_n), q_n) \notin \mathrm{dom}(\tau)$, then $(k_{n+1}, c_{n+1}, q_{n+1}) = (k_n, c_n, q_n)$.

(c) If $q_n \neq q_h$, $(c_n(k_n), q_n) \in \mathrm{dom}(\tau)$ and $\tau(c_n(k_n), q_n) = (s, q, j)$, then

 (i) $c_{n+1} \restriction (\mathbb{Z} \setminus \{k_n\}) = c_n \restriction (\mathbb{Z} \setminus \{k_n\})$ and $c_{n+1}(k_n) = s$,

 (ii) $q_{n+1} = q$ and

 (iii) $k_{n+1} = k_n + j$.

The triplet (k_n, c_n, q_n) describes the **machine configuration**. It stores the "head position", "tape configuration" and "state" of the TM at stage n of the run of the machine. Clause (a) is saying that the machines starts at tape position 0, tape configuration $c_\star$ and the start state q_s. Clause (b) is saying that if the current state is the halting state, then **the machine halts** and nothing changes in the future. If the transition function is undefined, then we say that **the machine stalls** (but not halts). Clause (c) is saying that if the current state is not the halting state and the transition function is defined at the current configuration then the tape head changes the symbol under the head to a new symbol, moves the tape head left/right and enters a new state as described by the transition function.

Definition 2.4. *Let $T = (\Sigma, S, B, q_s, q_h, \tau)$ be a TM. Put $\Gamma = \Sigma \setminus \{B\}$. Let $\sigma, \rho \in \Gamma^\star$. We say that on input σ, the machine T halts and outputs ρ iff for some $n < \omega$, the run of T on $c_\star$ at stage n equals (k_n, c_n, q_n) where*

- *$c_\star \restriction |\sigma| = \sigma$ and $c_\star(k) = B$ for all $k \notin dom(\sigma)$,*
- *q_n is the halting state and*
- *$c_n \restriction |\rho| = \rho$ and $c_n(k) = N$ for all $k \notin dom(\rho)$.*

***The partial function from $\Gamma^\star$ to $\Gamma^\star$ computed by** T, denoted f_T, is the set of all pairs (σ, ρ) such that T on input σ halts and outputs ρ.*

Let h be a unary partial function on ω. We say that h **is (partial) Turing computable** iff there is a Turing machine $T = (\Sigma, S, B, q_s, q_h, \tau)$ such that $\Sigma = \{1, B\}$ and for every $m, n < \omega$, $f(m) = n$ iff $f_T(\sigma) = \rho$ where $\sigma = 1^n$ and $\rho = 1^m$.

The notion of a Turing computable n-ary partial function on ω (for $n \geq 2$) can be defined as follows. Let $(x_1, x_2, \cdots, x_n) \mapsto \langle x_1, x_2, \cdots, x_n \rangle$ be a fixed primitive recursive bijection from ω^n to ω. Let g be an n-ary partial function on ω. Define h_g by $h_g(\langle x_1, x_2, \cdots, x_n \rangle) = g(x_1, x_2, \cdots, x_n)$. Then we say that g is an **n-ary partial Turing computable function** iff h_g is a unary partial Turing computable function.

3 Universal functions

Fact 3.1 (Kleene normal form)**.** *For each $k \geq 1$, there exists a primitive recursive $(k+2)$-ary function $P(e, z, x_1, \cdots, x_k)$ such that for every partial Turing computable k-ary function f, there exists $e < \omega$ such that*

(1) $dom(f) = \{(x_1, \cdots, x_k) \in \omega^k : (\exists z)(P(e, z, x_1, \cdots, x_k) = 0)\}$ and

(2) for every $(x_1, \cdots, x_k) \in dom(f)$,

$$f(x_1, \cdots, x_k) = \textsf{\textit{ the least } } z \textsf{\textit{ such that }} P(e, z, x_1, \cdots, x_k) = 0.$$

Proof idea. The function $P(e, z, \vec{x})$ is defined as follows. Write $e = \langle n, s, r\rangle$ and run the nth Turing machine on input $\vec{x}$ for s-stages. If r codes the run of the machine for s-stages and the machine configuration after stage s is in the halting state with output z, then $P(e, z, \vec{x}) = 0$. Otherwise, $P(e, z, \vec{x}) = 1$. We omit the somewhat tedious coding details that are needed to verify that P is indeed a PRec function. □

Fact 3.2. *A finitary partial function on ω is general recursive iff it is partial Turing computable.*

Proof idea. The fact that every partial Turing computable function is general recursive follows from Fact 3.1. The converse is established by designing Turing machines that compute the zero, projections and successor functions and showing that the set of partial Turing computable functions is closed under composition, primitive recursion and unbounded search. □

Besides Turing machines, there are many other "models of computation" like register machines, BASIC and C programs. The set of partial functions that are computable according to any of these models coincides with the set of partial Turing computable functions. In view of this, we don't have to commit ourselves to any fixed model of computation and we drop the adjective "Turing" from "Turing" computable.

The **Church-Turing thesis** says that every "intuitively partial computable function" is Turing computable. This is useful because on several occasions, to show that a given function is partial computable, we just describe an informal description of a "computer program" that can compute it. When challenged, we can always replace it by a Turing machine or a C-program etc. This is similar to how a math publication almost never provides an actual formal proof (say, in ZFC) but everyone agrees that such a proof can be provided.

Corollary 3.3 (Universal partial computable function)**.** *For each $k \geq 1$, define a $(k+1)$-ary partial computable function U_k by*

$$U_k(e, x_1, \cdots, x_k) = \textsf{\textit{ least }} z \textsf{\textit{ such that }} P(e, z, x_1, \cdots, x_k) = 0$$

where P is as in Fact 3.1. Then for every k-ary partial computable f, there exists $e < \omega$ such that

- *$dom(f) = \{(x_1, \cdots, x_k) \in \omega^k : (e, x_1, \cdots, x_k) \in dom(U_k)\}$ and*
- *for every $(x_1, \cdots, x_k) \in dom(f)$, $f(x_1, \cdots, x_k) = U_k(e, x_1, \cdots, x_k)$.*

We will write U instead of U_1 and *think of $U(e, x)$ as the (possibly non-existent) output when the eth computer program is run with input x.*

4 Indexing/Numbering

Convention. From now on, "a computable function" will mean "a total computable function".

Definition 4.1. *The eth partial computable function, denoted φ_e, is defined by $\varphi_e(x) = U(e, x)$. We say that $\varphi_e(x)$ converges (resp. diverges) and write $\varphi_e(x)\downarrow$ (resp. $\varphi_e(x)\uparrow$) iff $n \in dom(\varphi_e)$ (resp. $n \notin dom(\varphi_e)$).*

Lemma 4.2 (Padding)**.** *There is an injective computable function $h : \omega^2 \to \omega$ such that*

$$(\forall e, n < \omega)(\varphi_e = \varphi_{h(e,n)}).$$

Proof. Define $h(e, n)$ to be the index of the program obtained by adding $\langle e, n\rangle$-lines of harmless code (like declare a new variable and increment it $\langle e, n\rangle$ times) to the eth program. □

Lemma 4.3 (s^n_m theorem)**.** *Suppose $n, m \geq 1$ and $\theta(\vec{x}, \vec{y})$ is an $(n+m)$-ary partial computable function. Then there exists an injective computable $s : \omega^n \to \omega$ such that*

$$(\forall \vec{x} \in \omega^n)(\forall \vec{y} \in \omega^m)[\theta(\vec{x}, \vec{y}) = \varphi_{s(\vec{x})}(\langle y_1, \cdots, y_m\rangle)].$$

Proof. Fix a program P that computes θ. For each $\vec{x}$, define $g(\vec{x})$ to be the index of the program that on input $\langle y_1, \cdots, y_m\rangle$ runs P with input $(\vec{x}, \vec{y})$. Define $s(\vec{x}) = h(g(\vec{x}), \langle x_1, \cdots, x_n\rangle)$ (this is to ensure that s is injective) where h is the padding function of Lemma 4.2. $\square$

Theorem 4.4 (Kleene fixed-point/recursion theorem)**.** *For every computable $f : \omega \to \omega$, there are infinitely many e such that $\varphi_e = \varphi_{f(e)}$.*

Proof. Define $\theta(e, x) = \varphi_{\varphi_e(e)}(x)$ and note that θ is partial computable. Using s^1_1-theorem, fix an injective computable $s : \omega \to \omega$ such that $\theta(e, x) = \varphi_{s(e)}(x)$. Let i be any index for the computable function $f \circ s$ (so $\varphi_i = f \circ s$). Put $e = s(i)$. Then for every x,

$$\varphi_e(x) = \varphi_{h(i)}(x) = \theta(i, x) = \varphi_{\varphi_i(i)}(x) = \varphi_{f(s(i))}(x) = \varphi_e(x).$$

It follows that for every index i for $f \circ s$, $e = h(i)$ is as required. As h is injective and (by padding) there are infinitely many such i's, we get infinitely many e satisfying $\varphi_e = \varphi_{f(e)}$. $\square$

Quines are programs that on any input print their own source code. In our setup, eth program is a Quine iff φ_e is the (total) constant function e. Let us show that Quines exist. Define $\theta(e, x) = e$ for all $e, x < \omega$. By s^1_1-theorem, there is an injective computable $s : \omega \to \omega$ such that $(\forall e, x)(e = \theta(e, x) = \varphi_{s(e)}(x))$. So for every e, $\varphi_{h(e)}$ is the constant function e. By recursion theorem, there are infinitely many e such that $\varphi_e = \varphi_{s(e)} \equiv e$.

Exercise 4.5. *Show that there are infinitely many e such that dom$(\varphi_e) = \{e\}$.*

Definition 4.6 (Halting problem)**.** *The halting problem is the set $K = \{e < \omega : \varphi_e(e) \downarrow\}$.*

Definition 4.7. *$I \subseteq \omega$ is an index set iff for every e, e', if $\varphi_e = \varphi_{e'}$, then $(e \in I \iff e' \in I)$.*

Note that I is an index set iff $\omega \setminus I$ is an index set.

Exercise 4.8 (Joe Miller)**.** *Without using the recursion theorem, show that K is not an index set.*

Theorem 4.9 (Rice theorem)**.** *The only computable index sets are $\emptyset, \omega$.*

Proof. Let $I \notin \{\emptyset, \omega\}$ be an index set. Fix e' such that $\mathrm{dom}(\varphi_{e'}) = \emptyset$ and WLOG assume that $e' \notin I$ (otherwise replace I with $\omega \setminus I$). Fix $e'' \in I$. Define $\theta(e, x) = \varphi_{e''}(x)$ if $e \in K$ and undefined otherwise. Note that θ is partial computable. Using s^1_1-theorem fix an injective computable $s : \omega \to \omega$ such that $(\forall e, x)(\theta(e, x) = \varphi_{s(e)}(x))$. Now observe that

$$e \notin K \implies (\forall x)(\theta(e, x) \text{ is undefined}) \implies (\forall x)(\varphi_{s(e)}(x) \uparrow) \implies \varphi_{s(e)} = \varphi_{e'} \implies s(e) \notin I$$

and

$$e \in K \implies (\forall x)(\theta(e, x) = \varphi_{e''}(x)) \implies (\forall x)(\varphi_{s(e)}(x) = \varphi_{e''}(x)) \implies \varphi_{s(e)} = \varphi_{e''} \implies s(e) \in I$$

Suppose I is computable. Then K is also computable since the function s is computable and to check if $e \in K$, we can check if $s(e) \in I$. But K is not computable (Exercise 4.8). $\square$

5 C.e. sets

Definition 5.1. *$X \subseteq \omega$ is c.e. (computably enumerable) iff either $X = \emptyset$ or there is a computable function $f : \omega \to \omega$ such that range$(f) = X$.*

Theorem 5.2. *The following are equivalent for any $X \subseteq \omega$.*

(1) X is c.e.

(2) X is the range of some partial computable function.

(3) X is the domain of some partial computable function.

(4) For some $e < \omega$, $X = dom(\varphi_e)$.

(5) There exists a computable $R \subseteq \omega^2$ such that $X = \{y : (\exists x)((x, y) \in R)\}$.

Proof. (1) $\implies$ (2): Clear.

(2) $\implies$ (3): Fix a partial computable θ such that range$(\theta) = X$. Consider a program P that on input n starts computing $\theta(0), \theta(1), \cdots$ and halts as soon as it finds some k such that $\theta(k) \downarrow= n$. Let ψ be the partial computable function computed by P. Then dom$(\psi) = X$.

(3) $\implies$ (4): Clear.

(4) $\implies$ (5): Define R to be the set of all pairs $(x, y) \in \omega^2$ such that the eth Turing machine on input y halts in $\leq x$ steps.

(5) $\implies$ (1): Fix $R \subseteq \omega^2$ computable and define $X = \{n : (\exists m)((m, n) \in R)\}$. Assume $X \neq \emptyset$ and fix some $n_\star \in X$. Let $f : \omega \to \omega$ be defined as follows. $f(0) = n_\star$. $f(n+1)$ is the smallest $k \leq n$ such that $k \notin \{f(0), \cdots, f(n)\}$ and for some $m \leq n$, $(m, k) \in R$. If there is no such k, define $f(n+1) = n_\star$. Then f is computable and range$(f) = X$. □

Lemma 5.3. *$X \subseteq \omega$ is computable iff both X and $\omega \setminus X$ are c.e.*

Proof. If X is computable, then both $X, \omega \setminus X$ are computable and therefore c.e. Next assume that $X, \omega \setminus X$ are c.e. Fix programs P, Q such that for every n, P halts on input n iff $n \in X$ and Q halts on input n iff $n \in \omega \setminus X$. Consider a program R that on input n starts running both P and Q on input n and outputs 1 if P halts and 0 if Q halts. Then R computes the characteristic function of X. So X is computable. □

Lemma 5.4. *For every $f : \omega \to \omega$, the graph of f is c.e iff f is computable.*

Proof. If f is computable, then its graph is also computable since $(n, m) \in f$ iff $f(n) = m$. Now assume that the graph of f is c.e. and fix a program P such that for every $(n, m) \in \omega^2$, P halts on input (n, m) iff $f(n) = m$. Consider the program Q that on input n starts running P on inputs $(n, 0), (n, 1), \cdots$ and halts and outputs m as soon as P halts on input (n, m). Then Q computes f. Hence f is computable. □

Exercise 5.5. *Show that $K = \{e : \varphi_e(e) \downarrow\}$ and $K_0 = \{\langle e, n\rangle : \varphi_e(n) \downarrow\}$ are c.e. and not computable.*

Exercise 5.6. *Show that every infinite c.e. set has an infinite computable subset..*

Definition 5.7. *Let $A, B \subseteq \omega$. We say that A, B are computably separable iff there exists a computable $R \subseteq \omega$ such that $A \subseteq R$ and $B \subseteq (\omega \setminus R)$.*

Theorem 5.8. *There exists a pair of disjoint c.e. sets that are computably inseparable.*

Proof. Define $A = \{e : \varphi_e(e) \downarrow= 1\}$ and $B = \{e : \varphi_e(e) \downarrow= 0\}$. Observe that both A and B are c.e. Towards a contradiction, suppose $R \subseteq \omega$ is computable, $A \subseteq R$ and $B \cap R = \emptyset$. Then $h(n) = 1 - 1_R(n)$ is computable (here 1_R is the characteristic function of R). So there must be some $e_\star < \omega$ such that $h = \varphi_{e_\star}$. In particular, $\varphi_{e_\star}$ is total. Now check that $h(e_\star) \neq \varphi_{e_\star}(e_\star)$. A contradiction □

Exercise 5.9. *Show that there is a partial computable (unary) function that cannot be extended to a total computable function. Hint: Use Theorem 5.8.*

Definition 5.10. *$\langle A_s : s \in R\rangle$ is a uniformly computable sequence iff R is computable and the set*

$$\{(n, s) : s \in R \text{ and } n \in A_s\}$$

is computable.

Exercise 5.11. *Show that $A \subseteq \omega$ is c.e. iff there is a uniformly computable sequence $\langle A_s : s \in R\rangle$ such that $(\forall s)(A_s \subseteq A_{s+1})$ and $A = \bigcup_{s \in R} A_s$.*

Definition 5.12. *The eth c.e. set is defined by $W_e = dom(\varphi_e)$. Let $W_{e,s}$ be the set of all $n \leq s$ such that $\varphi_e(n)$ converges in $\leq s$ steps.*

Observe that $\langle W_{e,s} : e, s < \omega\rangle$ is a uniformly computable sequence of finite sets. Furthermore, $W_{e,s} \subseteq W_{e,s+1}$ and $W_e = \bigcup_{s<\omega} W_{e,s}$. We think of $W_{e,s}$ as the s-stage (finite) approximation to W_e.

6 Mapping reductions

Definition 6.1. *Let $A, B \subseteq \omega$. We say that A m-reduces to B and write $A \leq_m B$ iff there exists a computable $f : \omega \to \omega$ such that*

$$(\forall n)(n \in A \iff f(n) \in B).$$

Definition 6.2. *Let $A, B \subseteq \omega$. We say that A 1-reduces to B and write $A \leq_1 B$ iff there exists an injective computable $f : \omega \to \omega$ such that*

$$(\forall n)(n \in A \iff f(n) \in B).$$

Definition 6.3. *Define $A \equiv_m B$ iff $A \leq_m B$ and $B \leq_m A$. Similarly, $A \equiv_1 B$ iff $A \leq_1 B$ and $B \leq_1 A$.*

Show that $\leq_m$ and $\leq_1$ are both reflexive and transitive relations on $\mathcal{P}(\omega)$. Conclude that $\equiv_m$ and $\equiv_1$ are equivalence relations with countable equivalence classes. Also note that $A \leq_1 B$ implies $A \leq_m B$. Is the converse true?

Definition 6.4 (Joins)**.** *For $A, B \subseteq \omega$, define the join of A, B by*

$$A \oplus B = \{2n : n \in A\} \cup \{2n + 1 : n \in B\}$$

Exercise 6.5. *Show that $A \leq_1 A \oplus B$, $B \leq_1 A \oplus B$ and $A \equiv_m A \oplus A$.*

Exercise 6.6 (Joe Miller)**.** *Show that there is an infinite co-infinite set $A \subseteq \omega$ such that $A \oplus A \not\leq_1 A$.*

Exercise 6.7. *Assume $A \leq_m B$. Then B is computable (resp. c.e.) implies A is computable (resp. c.e.).*

Theorem 6.8. *Every c.e. set is 1-reducible to K.*

Proof. Let A be any c.e. set. Define $\theta(e, x) = 0$ if $e \in A$ and undefined otherwise. Note that θ is partial computable because A is c.e. Using s^1_1-theorem, fix an injective computable $s : \omega \to \omega$ such that $(\forall e, x)(\theta(e, x) = \varphi_{s(e)}(x))$. Now check that

$$e \in A \implies (\forall x)(\theta(e, x) = 0) \implies \varphi_{s(e)}(s(e)) \downarrow \implies s(e) \in K$$

and

$$e \notin A \implies (\forall x)((e, x) \notin \mathrm{dom}(\theta)) \implies \varphi_{s(e)}(s(e)) \uparrow \implies s(e) \notin K$$

So s is a 1-reduction of A to K. $\square$

Theorem 6.9 (Myhill)**.** *Suppose $A, B \subseteq \omega$ are such that $A \leq_1 B$ and $B \leq_1 A$. Show that there is a computable bijection $f : \omega \to \omega$ such that $f[A] = B$.*

Proof. Fix computable injections $g, h : \omega \to \omega$ such that for every x, $(x \in A \iff g(x) \in B)$ and $(x \in B \iff h(x) \in A)$. Define a uniformly computable sequence $\langle f_s : s < \omega \rangle$ of finite partial injective functions on ω as follows. Put $f_0 = \emptyset$. Suppose f_s has been defined such that $\mathrm{dom}(f_s)$ is finite and $(\forall x \in \mathrm{dom}(f_s))(x \in A \iff f_s(x) \in B)$. We consider two cases.

s **is even**: Choose the least x such that $x \notin \mathrm{dom}(f_s)$. We will find a $y \notin \mathrm{range}(h_s)$ such that $x \in A \iff y \in B$ and define $f_{s+1} = f_s \cup \{(x, y)\}$ as follows.

Put $x_0 = x$ and $y_0 = g(x_0)$. If $y_0 \notin \mathrm{range}(f_s)$, then $y = y_0$ works. Otherwise, put $x_1 = h_s^{-1}(y_0)$ and $y_1 = g(x_1)$. If $y_1 \notin \mathrm{range}(f_s)$, then define $y = y_1$ works. Otherwise, we continue in this fashion until we find some $y_k \notin \mathrm{range}(f_s)$ in which case we define $y = y_k$. This must happen at some stage because $y_0, y_1, \cdots$ are pairwise distinct and $\mathrm{range}(f_s)$ is finite so they cannot keep appearing in $\mathrm{range}(f_s)$. It is also easy to check that $x \in A \iff y_k \in A$ because both f_s and g map members/non-members of A to members/non-members of B.

s **is odd**: Choose the least y such that $y \notin \mathrm{range}(f_s)$ and repeat the argument for the even case with h instead of g to find an $x \notin \mathrm{dom}(f_s)$ such that $y \in B \iff x \in A$ and define $f_{s+1} = f_s \cup \{(x, y)\}$

Define $f = \bigcup_{s<\omega} f_s$. Note that the even stages ensure that $\mathrm{dom}(f) = \omega$ and the odd stages guarantee that $\mathrm{range}(f) = \omega$. So f is a bijection on ω and the graph of f is c.e. (by Exercise 5.11). Hence by Lemma 5.4, f is computable. $\square$

7 Immune and simple sets

Definition 7.1. *$X \subseteq \omega$ is immune iff X is infinite and it does not have any infinite c.e. (equivalenty, computable) subset.*

Exercise 7.2. *Let $\mathcal{F}$ be any countable family of infinite subsets of ω. Show that there is an infinite $X \subseteq \omega$ such that no set in $\mathcal{F}$ is a subset of X. Conclude that there are immune sets.*

By Exercise 5.6, a c.e. set cannot be immune. What about complements of c.e. sets?

Definition 7.3. *$E \subseteq \omega$ is a simple set iff E is c.e. and $\omega \setminus E$ is immune.*

Exercise 7.4. *Show that a c.e. set $E \subseteq \omega$ is simple iff $\omega \setminus E$ is infinite and for every e,*

$$W_e \text{ is infinite } \implies W_e \cap E \neq \emptyset.$$

Theorem 7.5 (Post)**.** *There is a simple set.*

Proof. Define a uniformly computable sequence $\langle E_s : s < \omega\rangle$ as follows. Define $E_0 = \emptyset$. At stage $s+1$, E_{s+1} is defined as follows. Search for the least $e < s$ such that $W_{e,s} \cap E_s = \emptyset$ and $(\exists x)(x > 2e$ and $x \in W_{e,s})$ and define $E_{s+1} = E_s \cup \{x\}$. If there is no such x, then $E_{s+1} = E_s$.

It is clear that $\langle E_s : s < \omega\rangle$ is uniformly computable. Therefore $E = \bigcup_{s<\omega} E_s$ is c.e. We claim that E is simple. First note that $(\forall e)(|E \cap \{0, 1, \cdots, 2e\}| \leq e)$ (Why?). Hence $\omega \setminus E$ is infinite.

Next, it suffices to show that for every e, if W_e is infinite, then $W_e \cap E \neq \emptyset$. Suppose this fails and fix the least e for which W_e is infinite and $W_e \cap E = \emptyset$. Now choose a stage $s > e$ such that the following hold.

(a) $(\exists x)(x > 2e$ and $x \in W_{e,s})$.

(b) For every $e' < e$, if $W_{e'} \cap E \neq \emptyset$, then $W_{e',s} \cap E_s \neq \emptyset$.

Then at stage $s+1$, we must have $E_{s+1} = E_s \cup \{x\}$ for some $x \in W_{e,s}$. A contradiction. □

Exercise 7.6. *Show that if $A \leq_1 E$, $\omega \setminus A$ is infinite and E is simple, then A is simple.*

Exercise 7.7. *Show that K is not simple and conclude that $K \not\leq_1 E$ for any simple set E.*

8 Oracles, Turing degrees and the jump operator

Definition 8.1. *An oracle Turing machine T consists of the following.*

(1) A finite alphabet Σ with a blank symbol $B \in \Sigma$.

(2) A finite set S whose members of S are called states of T.

(3) A start state $q_s \in S$ and a halting state $q_h \in S$.

(4) A transition function τ that is a partial function from $\Sigma \times \Sigma \times S$ to $\Sigma \times S \times \{-1, 1\} \times \{-1, 1\}$.

Let $T = (\Sigma, S, B, q_s, q_h, \tau)$ be an oracle TM. A **tape configuration** is a function $c : \mathbb{Z} \to \Sigma$ such that $\{n \in \mathbb{Z} : c(n) \neq B\}$ is finite. An **oracle** is a function $X : \mathbb{Z} \to \Sigma$. Given an initial tape configuration $c_\star : \mathbb{Z} \to \Sigma$, the **run of T with oracle X on $c_\star$** is a sequence $\langle (k_n, j_n, c_n, q_n) : n < \omega\rangle$ defined as follows.

(a) $(k_0, j_0, c_0, q_0) = (0, 0, c_\star, q_s)$.

(b) If either $q_n = q_h$ or $(c_n(k_n), X(j_n), q_n) \notin \mathrm{dom}(\tau)$, then $(k_{n+1}, j_{n+1}, c_{n+1}, q_{n+1}) = (k_n, j_n, c_n, q_n)$.

(c) If $q_n \neq q_h$, $(c_n(k_n), X(j_n), q_n) \in \mathrm{dom}(\tau)$ and $\tau(c_n(k_n), X(j_n), q_n) = (s, q, k, j)$, then

  (i) $c_{n+1} \restriction (\mathbb{Z} \setminus \{k_n\}) = c_n \restriction (\mathbb{Z} \setminus \{k_n\})$ and $c_{n+1}(k_n) = s$,

  (ii) $q_{n+1} = q$ and

  (iii) $k_{n+1} = k_n + k$ and $j_{n+1} = j_n + j$.

(k_n, j_n, c_n, q_n) describes the **machine configuration**. It stores the "work head position", "oracle head position", "tape configuration" and "state" of the TM at stage n of the run of the machine.

One can now go ahead and define what it means for a partial finitary function to be Turing computable from an oracle $X \subseteq \omega$ analogous to Definition 2.4. We skip the obvious details.

Definition 8.2. *Let $X \in 2^\omega$. The set of all general recursive functions in X, denoted $\mathsf{GRec}(X)$ is the smallest set of partial finitary functions on ω that contains X, all general recursive functions and is closed under compositions, primitive recursion and unbounded search.*

Fact 8.3. *For every partial finitary function f on ω and oracle $X \subseteq \omega$,*

$$f \text{ is Turing computable in } X \text{ iff } f \in \mathsf{GRec}(X).$$

Let $\langle P_e : e < \omega \rangle$ be a computable enumeration of all oracle machines with alphabet $\Sigma = \{1, B\}$. The eth **Turing functional** Φ_e is a partial function from $2^\omega \times \omega$ to ω defined as follows. For each $X \in 2^\omega$ and $n, m < \omega$, we write $\Phi_e^X(n) \downarrow = m$ iff P_e with oracle X on input n halts and outputs m. We write $\Phi_e^X(n) \uparrow$ iff P_e does not halt on input n.

Lemma 8.4 (Finite oracle use)**.** *Assume $\Phi_e^X(n) \downarrow$. There exists a finite $F \subseteq \omega$ such that for every $Y \in 2^\omega$,*

$$Y \restriction F = X \restriction F \implies \Phi_e^X(n) \downarrow = \Phi_e^Y(n) \downarrow .$$

For any such finite $F \subseteq \omega$, we call $X \restriction F$ an ***oracle use of the computation*** *$\Phi_e^X(n)$. We will sometimes write $\Phi_e^{X \restriction F}(n) \downarrow$ etc. in this case.*

Definition 8.5. *Suppose $\sigma \in {}^{<\omega}2$ and $e, n < \omega$. We write $\Phi_e^\sigma(n) = m$ iff for some $X \in 2^\omega$ such that $\sigma \subseteq X$ and $\Phi_e^X(n)$ and σ is a use of this computation.*

Definition 8.6 (Turing reduction)**.** *Let $X, Y \in 2^\omega$. Define $X \leq_T Y$ iff there exists $e < \omega$ such that $(\forall n < \omega)(\Phi_e^Y(n) = X(n))$.*

The oracle analogues of padding lemma, s_m^n-theorem and Kleene recursion theorem are as follows. Their proofs are similar.

Lemma 8.7 (Padding)**.** *There is an injective computable function $h : \omega^2 \to \omega$ such that*

$$(\forall X \in 2^\omega)(\forall e, n < \omega)(\Phi_e^X = \Phi_{h(e,n)}^X).$$

Lemma 8.8 (s_m^n-theorem)**.** *Suppose $n, m \geq 1$, $X \in 2^\omega$ and $\theta(\vec{x}, \vec{y})$ is an $(n+m)$-ary partial computable function in X. Then there exists an injective computable $s : \omega^n \to \omega$ such that*

$$(\forall \vec{x} \in \omega^n)(\forall \vec{y} \in \omega^m)[\theta(\vec{x}, \vec{y}) = \Phi_{s(\vec{x})}^X(\langle y_1, \cdots, y_m \rangle)].$$

Lemma 8.9 (Fixed point)**.** *For every computable $f : \omega \to \omega$, there are infinitely many e such that*

$$(\forall X \in 2^\omega)(\Phi_e^X = \Phi_{f(e)}^X).$$

It is easy to see that $\leq_T$ is a reflexive and transitive relation on 2^ω. Define $X \equiv_T Y$ iff $X \leq_T Y$ and $Y \leq_T X$. Then $\equiv_T$ is an equivalence relation on 2^ω. Then $\{Y \in 2^\omega : X \equiv_T Y\}$ is the $\equiv_T$-equivalence class of X and is countable. Let $\mathcal{D}$ be the set of all $\equiv_T$-equivalence classes. Members of $\mathcal{D}$ are called **Turing degrees**. For $\mathbf{a}, \mathbf{b} \in \mathcal{D}$, we write $\mathbf{a} \leq_T \mathbf{b}$ iff $X \leq_T Y$ for every $X \in \mathbf{a}$ and $Y \in \mathbf{b}$. $\mathbf{0}$ is the Turing degree of any computable set.

Exercise 8.10. *$X \leq_m Y$ implies $X \leq_T Y$.*

Definition 8.11 (Turing jump)**.** *For each $X \in 2^\omega$, define the (Turing) jump of X by*

$$X' = \{e < \omega : \Phi_e^X(e) \downarrow\}.$$

Lemma 8.12. *The following hold for all $X, Y \in 2^\omega$.*

(1) $X \leq_1 X'$ *and* $X \leq_T X'$.

(2) $X' \not\leq_T X$.

(3) $X \leq_T Y \implies X' \leq_1 Y'$.

(4) $X \equiv_T Y \implies X' \equiv_T Y'$.

Proof. (1) Define $\theta(e, n) = 0$ if $e \in X$ and undefined otherwise. Then θ is a partial computable function in X. By the s^1_1-theorem, there is an injective computable $h : \omega \to \omega$ such that for every $e, n < \omega$, $\theta(e, n) = \Phi^X_{h(e)}(n)$. It follows that $e \in X$ iff $\Phi^X_{h(e)}(h(e)) \downarrow$ iff $h(e) \in X'$. So $X \leq_1 X'$. That $X \leq_T X'$ follows from Exercise 8.10.

(2) Exercise.

(3) Assume $X \leq_T Y$. Define $\theta(e, n) = \Phi^X_e(e)$. Since $X \leq_T Y$, if follows that θ is partial computable in Y. By the s^1_1-theorem, there is an injective computable $h : \omega \to \omega$ such that for every $e, n < \omega$, $\theta(e, n) = \Phi^Y_{h(e)}(n)$. It follows that $e \in X'$ iff $(\forall n)(\theta(e, n) \downarrow)$ iff $\Phi^Y_{h(e)}(h(e)) \downarrow$ iff $h(e) \in Y'$. So $X' \leq_1 Y'$.

(4) Follows from (3).

□

Definition 8.13 (Jump of a degree)**.** *The jump of a Turing degree* $\mathbf{a} \in \mathcal{D}$ *(denoted by* $\mathbf{a}'$*) is the Turing degree of* X' *for any* $X \in \mathbf{a}\}$*. This is well-defined by By Lemma 8.12(4). For each* $n \geq 1$*,* $\mathbf{0}^n$ *is defined by* $\mathbf{0}^1 = \mathbf{0}'$ *and* $\mathbf{0}^{n+1} = (\mathbf{0}^n)'$*.*

Exercise 8.14. *Show that* $\mathbf{0}'$ *is the degree of the halting problem* K*.*

Recall that for $X, Y \in 2^\omega$, the join of X, Y is defined by $X \oplus Y = \{\langle m, n\rangle : X(m) = 1, Y(n) = 1\}$ (Definition 6.4). It is easy to see that if $X_1 \equiv_T X_2$ and $Y_1 \equiv_T Y_2$, then $X_1 \oplus Y_1 \equiv_T X_2 \oplus Y_2$.

Definition 8.15 (Join of two degrees)**.** *For* $\mathbf{a}, \mathbf{b} \in \mathcal{D}$*, define* $\mathbf{a} \oplus \mathbf{b}$*, called the join of* $\mathbf{a}$ *and* $\mathbf{b}$*, to be the Turing degree of* $X \oplus Y$ *where* $X \in \mathbf{a}$ *and* $Y \in \mathbf{b}$*.*

Exercise 8.16. *Show that* $\mathbf{a} \oplus \mathbf{b}$ *is the* $\leq_T$*-least upper bound of* $\{\mathbf{a}, \mathbf{b}\}$ *in* $\mathcal{D}$*.*

9 Incomparable degrees

Lemma 9.1. *The poset* $(\mathcal{D}, \leq_T)$ *satisfies the following.*

(1) $|\mathcal{D}| = |2^\omega| = \mathfrak{c}$*.*

(2) $\mathbf{0}$ *is the* $\leq_T$*-least member of* $\mathcal{D}$*.*

(3) Every finite subset of $\mathcal{D}$ *has a* $\leq_T$*-least upper bound.*

(4) For every $\mathbf{a} \in \mathcal{D}$*, there exists* $\mathbf{b} \in \mathcal{D}$ *such that* $\mathbf{a} <_T \mathbf{b}$*.*

(5) Every countable subset of $\mathcal{D}$ *has a* $\leq_T$*-upper bound.*

Proof. Each $\mathcal{D}$ is a partition of the uncountable set 2^ω and each $\mathbf{a} \in \mathcal{D}$ has size ω. Therefore (1) holds. (2) is trivial. (3) follows from Exercise 8.16. For (4), take $\mathbf{b} = \mathbf{a}'$.

Finally, assume $\{\mathbf{a}_n : n < \omega\} \subseteq \mathcal{D}$. For each $n < \omega$, fix some $X_n \in \mathbf{a}_n$ and define $X = \{\langle n, k\rangle : X_n(k) = 1\}$. Then each $X_n \leq_T X$ and therefore the Turing degree of X is an upper bound of $\{\mathbf{a}_n : n < \omega\}$. So (5) holds. □

We will next address the question: Is $\leq_T$ a linear order on $\mathcal{D}$?

Exercise 9.2. *Define a distance function on* 2^ω *by* $d(X, X) = 0$ *and* $d(X, Y) = 2^{-\Delta(X,Y)}$ *if* $X \neq Y$ *where* $\Delta(X, Y) = \min(\{n : X(n) \neq Y(n)\})$*.*

(1) Show that $(2^\omega, d)$ is a compact metric space.

(2) For a finite $F \subseteq \omega$ and $\sigma : F \to 2$, define $[\sigma] = \{X \in 2^\omega : \sigma \subseteq X\}$. Show that the family $\{[\sigma] : \sigma \in {}^{<\omega}2\}$ is a clopen basis of $(2^\omega, d)$.

Let $\mathcal{B}$ be the σ-algebra generated by the family of all open sets in 2^ω. Member of $\mathcal{B}$ are called **Borel subsets of** 2^ω. There is a unique probability measure $\mu : \mathcal{B} \to [0, 1]$ (called **Lebesgue/fair coin-toss/Bernoulii measure**) that satisfies $\mu([\sigma]) = 2^{-|\sigma|}$ for every $\sigma \in {}^{<\omega}2$.

Exercise 9.3 (Lebesgue density)**.** *Suppose $E \subseteq 2^\omega$ is Borel and $\mu(E) > 0$. Then for each $\varepsilon > 0$, there exists $\sigma \in {}^{<\omega}2$ such that $\mu(E \cap [\sigma]) > (1 - \varepsilon)\mu([\sigma])$.*

Exercise 9.4 (Inner regularity)**.** *For every Borel $E \subseteq 2^\omega$,*

$$\mu(E) = \sup\left(\{\mu(K) : K \subseteq E \textit{ and } K \textit{ is compact}\}\right).$$

Let $X \subseteq 2^\omega$. We say that X is **nowhere dense in** 2^ω iff for every nonempty open $U \subseteq 2^\omega$, there exists nonempty open $V \subseteq 2^\omega$ such that $X \cap V = \emptyset$. X is **meager** in 2^ω iff there is a countable family $\{A_n : n < \omega\}$ where each A_n is a nowhere dense subsets of 2^ω such that $X \subseteq \bigcup_{n<\omega} A_n$. The **Baire category theorem** implies that every non-empty open subset of 2^ω is non-meager in 2^ω.

Exercise 9.5. *Suppose $X \subseteq 2^\omega$ is non-meager. Show that for some $\sigma \in {}^{<\omega}2$, $X \cap [\sigma]$ is dense in $[\sigma]$.*

Lemma 9.6. *For each $X \in 2^\omega$ and $e < \omega$, $\{Y \in 2^\omega : \Phi_e^Y = X\}$ and $\{Y \in 2^\omega : X \leq_T Y\}$ are both Borel subsets of 2^ω.*

Proof. Put $A_e = \{Y \in 2^\omega : \Phi_e^Y = X\}$. Note that for each $n < \omega$, $V_{e,n} = \{Y \in 2^\omega : \Phi_e^Y(n) = X(n)\}$ is an open subset of 2^ω (see Lemma 8.4). Hence $Y \in A_e$ iff $(\forall n < \omega)(Y \in V_{e,n})$. So $A_e = \bigcap_{n<\omega} V_{e,n}$ is a countable union of open sets and hence Borel. Since $\{Y \in 2^\omega : X \leq_T Y\} = \bigcup_{e<\omega} A_e$, it is also Borel. □

Lemma 9.7 (Sacks)**.** *Suppose $X \in 2^\omega$ is non-computable. Then $\{Y \in 2^\omega : X \leq_T Y\}$ is both meager and has measure zero.*

Proof. Put $A = \{Y \in 2^\omega : X \leq_T Y\}$ and $A_e = \{Y \in 2^\omega : \Phi_e^Y = X\}$. Then $A = \bigcup_{e<\omega} A_e$. It is enough to show that for every $e < \omega$, A_e is both meager and has measure zero. Both A_e, A are Borel by Lemma 9.6.

First assume $\mu(A_e) > 0$ and we will show that X is computable which is a contradiction. By Exercise 9.3, we can find $\sigma \in {}^{<\omega}2$ such that $\mu(A_e \cap [\sigma]) > 0.9\mu([\sigma])$.

Consider the following program P that on input n searches for a finite set $\{\tau_k : k < N\} \subseteq {}^{<\omega}2$ and $j \in \{0, 1\}$ such that

- $[\tau_k]$'s are pairwise disjoint,
- $\displaystyle\sum_{k<N} \mu([\tau_k]) > 0.5\mu([\sigma])$ and
- $\Phi_e^{\tau_k}(n) \downarrow= j$.

and outputs j.

Exercise 9.8. *Show that this search is successful and P computes X.*

Now assume that A_e is non-meager. Once again, we'll show that X is computable. By Exercise 9.5, we can fix some $\sigma \in {}^{<\omega}2$ such that $A_e \cap [\sigma]$ is dense in $[\sigma]$. Consider the program Q that on input n, searches for some $\tau \in {}^{<\omega}2$ such that $\sigma \subseteq \tau$ and $\Phi_e^\tau(n) \downarrow= j$ and outputs j. Using the fact that $A_e \cap [\sigma]$ is dense in $[\sigma]$, it is easy to see that this search is successful and $X(n) = j$. □

Corollary 9.9. *For every non-computable $X \in 2^\omega$, there exists $Y \in 2^\omega$ such that X, Y are $\leq_T$-incomparable.*

Proof. The set $\{Y : Y \leq_T X\}$ is countable and therefore is both meager and has measure zero. Also, by Lemma 9.7, $\{Y : X \leq_T Y\}$ is both meager and has measure zero. The claim follows. □

Exercise 9.10. *Let $\nu = \mu \otimes \mu$ be the product measure on $2^\omega \times 2^\omega$. Show that $\{(X,Y) \in 2^\omega \times 2^\omega : X \leq_T Y\}$ is both ν-null and meager in $2^\omega \times 2^\omega$.*

Theorem 9.11 (Kleene-Post)**.** *There exist incomparable Turing degrees $\mathbf{a}, \mathbf{b} <_T \mathbf{0}'$.*

Proof. Fix a computable enumeration of ${}^{<\omega}2$ and inductively construct $\langle \sigma_n, \tau_n : n < \omega \rangle$ as follows.

(a) $\sigma_0 = \tau_0 = \emptyset$.

(b) Given σ_n, τ_n, define σ_{n+1} and τ_{n+1} as follows. Put $\ell = |\sigma_n|$ and $k = |\tau_n|$.

First suppose $n = 2e$ is even. We ask the following:

Is there some $\rho \in {}^{<\omega}2$ such that $\sigma_n \prec \rho$ and $\Phi_e^\rho(k)\downarrow$?

If the answer is yes, we choose the first such ρ, choose $j \in \{0,1\}$ such that $j \neq \Phi_e^\rho(k)$, and define $\sigma_{n+1} = \rho$ and $\tau_{n+1} = \tau_n \cup \{(k,j)\}$. If the answer is no, then we define $\sigma_{n+1} = \sigma_n \cup \{(\ell, 0)\}$ and $\tau_{n+1} = \tau_n \cup \{(k,0)\}$.

If $n = 2e+1$ is odd, we repeat the previous case with the roles of σ_n and τ_n reversed.

Put $X = \bigcup_{n,\omega} \sigma_n$ and $Y = \bigcup_{n<\omega} \tau_n$.

Exercise 9.12. *Show that X and Y are Turing incomparable.*

Note that the set $\{(e, \sigma, k) : (\exists \rho \in {}^{<\omega}2)(\sigma \preceq \rho \text{ and } \Phi_e^\rho(k)\downarrow)\}$ is c.e. and therefore computable in K. Therefore, the sequence $\langle (\sigma_n, \tau_n) : n < \omega \rangle$ is uniformly computable in K. It follows that $X, Y <_T K$. $\square$

Definition 9.13. *A c.e. degree is the Turing degree of a c.e. set.*

10 Turing independence

For a finite $F = \{x_0, x_1, \cdots, x_n\} \subseteq 2^\omega$, define the Turing join of F by $\bigoplus_{i<n} x_i = x \in 2^\omega$ where $x(nj+k) = x_k(j)$ for every $k < n$ and $j < \omega$.

Definition 10.1. *$X \subseteq 2^\omega$ is Turing independent iff for every finite $F \subseteq X$ and $y \in X \setminus F$, y is not computable from the Turing join of F.*

Definition 10.2. *$X \subseteq 2^\omega$ is maximal Turing independent iff X is Turing independent and for every Turing independent $Y \supseteq X$, $Y = X$.*

Exercise 10.3. *Use Zorn's lemma to show that Turing independent set can be extended to a maximal Turing independent set.*

Exercise 10.4. *Generalize Lemma 9.7 to show that for every $x, y \in 2^\omega$, if $x \not\leq_T y$, then $\{z \in 2^\omega : x \leq_T y \oplus z\}$ is both meager and has measure zero. Use this to show that every maximal Turing independent set is uncountable.*

Definition 10.5. *$T \subseteq {}^{<\omega}2$ is a **tree** iff for every $\sigma \in T$ and $\tau \in {}^{<\omega}2$, if $\tau \preceq \sigma$, then $\tau \in T$. For a tree $T \subseteq {}^{<\omega}2$, define the set of branches through T, by*

$$[T] = \{x \in 2^\omega : (\forall n < \omega)(x \upharpoonright n \in T)\}.$$

Members of a tree $T \subseteq {}^{<\omega}2$ will sometimes be referred to as **nodes** in T. If $\sigma \in T$ has no proper extension in T, we say that σ is a **leaf/terminal node** in T. Note that a finite tree is completely determined by the set of its terminal nodes.

Exercise 10.6. *For $X \subseteq 2^\omega$, define $T_X = \{y \upharpoonright k : y \in X \text{ and } k < \omega\}$. Show that $T_X \subseteq {}^{<\omega}2$ is a leafless tree and $[T_X]$ is the closure of X in 2^ω. Conclude that $X \subseteq 2^\omega$ is closed in 2^ω iff there exists a tree $T \subseteq {}^{<\omega}2$ such that $[T] = X$.*

For $\sigma \in {}^{n}2$ and $k < 2$, define $\sigma^\frown k = \sigma \cup \{(n,k)\}$. If $T \subseteq {}^{<\omega}2$ is a tree $\sigma \in T$ and both $\sigma^\frown 0, \sigma^\frown 1 \in T$, then we say that σ is a **splitting node** of T.

Definition 10.7. *A tree $T \subseteq {}^{<\omega}2$ is perfect iff for every $\sigma \in T$, there exists $\tau \in T$ such that $\sigma \preceq \tau$ and $\tau^\frown 0$ and $\tau^\frown 1$ are both in T.*

Let (X, d) be a metric space and $P \subseteq X$. Recall that a P is a **perfect subset of** X iff P is closed in X and P has no isolated points (for every $y \in P$, and $r > 0$, there exists $x \in P$ such that $0 < d(x, y) < r$).

Exercise 10.8. *Let $P \subseteq 2^\omega$. Show that P is perfect subset of 2^ω iff there is a perfect tree $T \subseteq {}^{<\omega}2$ such that $P = [T]$.*

Theorem 10.9 (Sacks)**.** *For every perfect tree $T \subseteq {}^{<\omega}2$, there exists a perfect tree $S \subseteq {}^{<\omega}2$ such that $S \subseteq T$ and $[S]$ is Turing independent.*

The inductive constructions involved in the proof of Theorem 10.9 and several later results share many common features and are referred to as forcing constructions. The following notions will be useful to to describe these common features.

10.1 Forcing

Definition 10.10. *A partial ordering/poset/forcing notion is a pair $(\mathbb{P}, \leq_{\mathbb{P}})$ such that $\leq_{\mathbb{P}}$ is a binary relation on $\mathbb{P}$ that satisfies the following.*

(a) (Reflexive) For all $p \in \mathbb{P}$, $p \leq_{\mathbb{P}} p$.

(b) (Transitive) For all $p, q, r \in \mathbb{P}$, if $p \leq_{\mathbb{P}} q$ and $q \leq_{\mathbb{P}} r$, then $q \leq_{\mathbb{P}} r$.

Note that we do not require anti-symmetry $p \leq_{\mathbb{P}} q \wedge q \leq_{\mathbb{P}} p \implies p = q$. We sometimes refer to members of $\mathbb{P}$ as conditions read $p \leq_{\mathbb{P}} q$ by "the condition p extends q" or "p is a stronger condition than q". Some examples of forcing notions follow.

Example 10.11. *(1) $\mathbb{P} = {}^{<\omega}2$ ordered by $\sigma \leq_{\mathbb{P}} \tau$ iff $\sigma \preceq \tau$. This is called **Cohen forcing**.*

*(2) $\mathbb{P}$ is the set of all perfect trees $T \subseteq {}^{<\omega}2$ ordered by $T \leq_{\mathbb{P}} S$ iff $T \subseteq S$. This is called **Sacks forcing**.*

*(3) $\mathbb{P}$ is the set of all perfect trees $T \subseteq {}^{<\omega}2$ such that $(\forall x \in [T])(T \leq_T x)$ (every branch of T computes T). Members of $\mathbb{P}$ are called recursively pointed perfect trees. The ordering is $T \leq_{\mathbb{P}} S$ iff $T \subseteq S$. We call $(\mathbb{P}, \leq_{\mathbb{P}})$ **forcing with recursively pointed perfect trees**.*

Following standard abuses of notation, we'll sometimes write "$\mathbb{P}$ is a poset" instead of "$(\mathbb{P}, \leq_{\mathbb{P}})$ is poset" when the ordering is clear from the context.

Definition 10.12 (Filter on a poset)**.** *Let $\mathbb{P}$ be a poset. A filter on $\mathbb{P}$ is a nonempty subset $G \subseteq \mathbb{P}$ satisfying the following.*

(i) For every $p, q \in G$, there exists $r \in G$ such that $r \leq_{\mathbb{P}} p$ and $r \leq_{\mathbb{P}} q$.

(ii) For every $p \in G$ and $q \in P$, if $p \leq_{\mathbb{P}} q$, then $q \in G$.

Definition 10.13 (Compatible, Dense)**.** *Let $\mathbb{P}$ be a poset.*

(a) We say that $p, q \in \mathbb{P}$ are compatible, iff p, q have a common extension in P; i.e., there exists $r \in \mathbb{P}$ such that $r \leq_{\mathbb{P}} p$ and $r \leq_{\mathbb{P}} q$. We say that p, q are incompatible, denoted $p \perp_{\mathbb{P}} q$, iff they are not compatible.

(b) A subset $D \subseteq \mathbb{P}$ is dense in $\mathbb{P}$ iff for every $p \in \mathbb{P}$, there exists $q \in D$ such that $q \leq_{\mathbb{P}} p$.

Lemma 10.14. *Suppose $\mathbb{P}$ is poset and $\mathcal{F}$ is a countable family of dense subsets of $\mathbb{P}$. Then there is a filter G on $\mathbb{P}$ such that for every $D \in \mathcal{F}$, $G \cap D \neq 0$.*

Proof. Let $\langle D_n : n < \omega \rangle$ enumerate $\mathcal{F}$. Inductively construct $\langle p_n : n < \omega \rangle$ as follows. $p_0 \in D_0$ is arbitrary. Suppose $p_0 \geq_{\mathbb{P}} p_1 \geq_{\mathbb{P}} \cdots \geq_{\mathbb{P}} p_n$ have been chosen such that for every $k \leq n$, $p_k \in D_k$. Since D_{n+1} is dense in $\mathbb{P}$, we can find $p_{n+1} \in D_{n+1}$ such that $p_{n+1} \leq_{\mathbb{P}} p_n$.

Define $G = \{q \in \mathbb{P} : (\exists n < \omega)(p_n \leq_{\mathbb{P}} q)\}$. Then it is easy to check that G is a filter on $\mathbb{P}$ that meets every D_n. ☐

Proof of Theorem 10.9. Fix a perfect tree T. Define a forcing $\mathbb{P}$ as follows. $p \in \mathbb{P}$ iff $p \subseteq T$ is a finite tree. For $p, q \in \mathbb{P}$, define $q \leq_{\mathbb{P}} p$ iff $p \subseteq q$ and for every terminal node τ in q, there is a terminal node σ in p such that $\sigma \preceq \tau$.

For each $e, n < \omega$, let $D_{e,n}$ consist of those $p \in \mathbb{P}$ which satisfy the following.

(i) Every terminal node in p has length $\geq n$.

(ii) For every injective sequence $\overline{\sigma} = \langle \sigma_k : k \leq N+1 \rangle$ of terminal nodes in p, if there exists $\langle x_k : k \leq N \rangle$ such that each $\sigma_k \preceq x_k \in [T]$ and Φ_e^X is total where $X = \bigoplus_{k \leq N} x_k$, then there exists $m \in \mathrm{dom}(\sigma_{N+1})$ such that $\Phi_e^\rho(m) \downarrow \neq \sigma_{N+1}(m)$ where $\rho = \bigoplus_{k \leq N} \sigma_k$.

Let E_n be the set of $p \in \mathbb{P}$ such that for every terminal node $\sigma \in p$,

$$|\{\tau \in p : \tau \preceq \sigma \text{ and } \tau \text{ is a splitting node in } p\}| \geq n.$$

Claim 10.15. *For every $e, n < \omega$, $D_{e,n}$ and E_n are both dense in $\mathbb{P}$.*

Proof. That E_n is dense follows from the fact that T is a perfect tree and is left as an exercise. Next, fix e, n and $p \in \mathbb{P}$. We will find an extension $q \in \mathbb{P}$ of p such that $q \in D_{e,n}$.

Definition 10.16. *Suppose $p \in \mathbb{P}$ and $\overline{\sigma} = \langle \sigma_k : k \leq N+1 \rangle$ is a finite sequence of terminal nodes in p and $e < \omega$. We write $\mathsf{Split}_e(\overline{\sigma}, p, T)$ for the following statement. For every $\langle x_k : k \leq N \rangle$ where each $\sigma_k \preceq x_k \in [T]$, there exists $\ell < |\sigma_{N+1}|$ such that letting $X = \bigoplus_{k \leq N} x_k$,*

(A) either $\Phi_e^X(\ell) \uparrow$ or

(B) $\Phi_e^X(\ell) \downarrow \neq \sigma_{N+1}(\ell)$ and the oracle use of the computation $\Phi_e^X(\ell) \downarrow$ is contained in $\bigoplus_{k \leq N} \sigma_k$.

Lemma 10.17. *Suppose $p \in \mathbb{P}$, $\overline{\sigma} = \langle \sigma_k : k \leq N+1 \rangle$ is a finite sequence of terminal nodes in p, $e < \omega$ and $\mathsf{Split}_e(\overline{\sigma}, p, T)$ holds. Let $q \in \mathbb{P}$, $q \leq p$ and $\overline{\tau} = \langle \tau_k : k \leq N+1 \rangle$ be a finite sequence of terminal nodes in p such that $\sigma_k \preceq \tau_k$ for each $k \leq N+1$. Then $\mathsf{Split}_e(\overline{\tau}, q, T)$ holds.*

Proof. Left to the reader. ☐

Definition 10.18. *Suppose $p \in \mathbb{P}$ and $e < \omega$. We say that $\mathsf{Split}_e(p, T)$ holds iff for every finite sequence $\overline{\sigma} = \langle \sigma_k : k \leq N+1 \rangle$ of terminal nodes in p, $\mathsf{Split}_e(\overline{\sigma}, p, T)$ holds.*

Lemma 10.19. *For every $p \in \mathbb{P}$ and $e < \omega$, there exists $q \in \mathbb{P}$ such that $\mathsf{Split}_e(q, T)$ holds and*

$$(\forall \sigma \in q \setminus p)(\exists \tau \in p)(\tau \text{ is a terminal node of } p \text{ and } \tau \preceq \sigma).$$

Proof. Done in class. ☐

By Lemma 10.14, there is a filter G on $\mathbb{P}$ such that for every $e, n < \omega$, $G \cap D_{e,n} \neq \emptyset$ and $G \cap E_n \neq \emptyset$. Put $\bigcup G = S$. Now check that $S \subseteq T$ is a perfect tree and $[S]$ is Turing independent. ☐

Definition 10.20. $\mathbb{P}$ *is a* ***locally finite/countable poset*** *iff the set of predecessors of every member of $\mathbb{P}$ is finite/countable.*

Exercise 10.21. *Let $\mathbb{P}$ be a locally finite poset of size $|\mathbb{P}| \leq |\mathbb{R}| = \mathfrak{c}$. Show that $\mathbb{P}$ is isomorphic to a subordering of the Turing degrees $(\mathcal{D}, \leq_T)$.*

Question 10.22 (Sacks)**.** *Is every locally countable poset isomorphic to a subordering of the Turing degrees?*

11 Exact pair

Theorem 11.1 (Spector)**.** *Let $\langle x_n : n < \omega\rangle$ be a $\le_T$-increasing sequence in 2^ω. Then there exist $a \neq b$ in 2^ω such that*

$$\{y \in 2^\omega : y \le_T a \text{ and } y \le_T b\} = \{y \in 2^\omega : (\exists n)(y \le_T x_n)\}.$$

We call any such pair $\{a, b\}$ an exact pair for $\{x_n : n < \omega\}$.

Proof. Define a forcing $\mathbb{P}$ as follows. $p \in \mathbb{P}$ iff $p = (K_p, p_0, p_1)$ where $K_p < \omega$, $p_0, p_1 : K_p \times \omega \to 2$ and for every $k < K$ and $i < 2$, $\{n < \omega : p_i(k, n) \neq x_k(n)\}$ is finite. For $p, q \in \mathbb{P}$ define $p \le_{\mathbb{P}} q$ iff $K_q \le K_p$ and $q_i \subseteq p_i$ for each $i < 2$.

Lemma 11.2. *Let $D_{e,e'}$ be the set of all $p \in \mathbb{P}$ such that the following hold. If there are $x_0, x_1 : \omega \times \omega \to 2$ and $\ell < \omega$ such that $p_0 \subseteq x_0$, $p_1 \subseteq x_1$ and $\Phi^{x_0}_{e_0}(\ell)\downarrow \neq \Phi^{x_1}_{e_1}(\ell)\downarrow$, then there exists $k < \omega$ such that $\Phi^{p_0}_{e_0}(k)\downarrow \neq \Phi^{p_1}_{e_1}(k)\downarrow$. Then $D_{e,e'}$ is dense in $\mathbb{P}$.*

Proof. Done in lecture. □

Exercise 11.3. *Let $E_n = \{p \in \mathbb{P} : K_p \ge n\}$. Then E_n is dense in $\mathbb{P}$ for each $n < \omega$.*

By Lemma 10.14, there is a filter G on $\mathbb{P}$ such that for every $e, e', n < \omega$, $G \cap D_{e,e'} \neq \emptyset$ and $G \cap E_n \neq \emptyset$. Put $a = \bigcup\{p_0 : p \in G\}$ and $b = \bigcup\{p_1 : p \in G\}$.

Claim 11.4. $\{y \in 2^\omega : y \le_T a$ *and* $y \le_T b\} = \{y \in 2^\omega : (\exists n)(y \le_T x_n)\}$.

Proof. For each $k < \omega$, $\{j < \omega : a(k, j) \neq x_k(j)\}$ and $\{j < \omega : b(k, j) \neq x_k(j)\}$ are both finite. So $x_k \le_T a$ and $x_k \le b$. It follows that

$$\{y \in 2^\omega : (\exists n)(y \le_T x_n)\} \subseteq \{y \in 2^\omega : y \le_T a \text{ and } y \le_T b\}.$$

Next suppose $y \le_T a$ and $y \le_T b$ and fix $e, e' < \omega$ such that $\Phi^a_e = \Phi^b_{e'} = y$. Fix $p \in D_{e,e'} \cap G$. Then $p_0 \subseteq a$ and $p_1 \subseteq b$ and $p_0 \equiv_T p_1 \le_T x_{K_p}$. So it suffices to show that $y \le_T p_0$.

Consider the oracle program Q using p_0 as an oracle that on input k, searches for a finite $F \subseteq (\omega \setminus K_p) \times \omega$ and $\rho : F \to 2$ such that $\Phi^{p_0 \cup \rho}_e(k)\downarrow$ and outputs $\Phi^{p_0 \cup \rho}_e(k)$.

We claim that Q halts on every input k and outputs $y(k)$. That it halts follows from that fact that $\Phi^a_e(k)\downarrow$ and $p_0 \subseteq a$. To see that it correctly outputs $y(k)$, towards a contradiction, suppose not and fix $F \subseteq (\omega \setminus K_p) \times \omega$ finite and $\rho : F \to 2$ such that $\Phi^{p_0 \cup \rho}_e(k)\downarrow \neq y(k) = \Phi^a_e(k)\downarrow$. Then since $p \in D_{e,e'}$, we get that for some $\ell < \omega$, $\Phi^{p_0}_e(\ell)\downarrow \neq \Phi^{p_1}_{e'}(\ell)\downarrow$. But $p_0 \subseteq a$, $p_1 \subseteq b$ and $\Phi^a_e(\ell)\downarrow = \Phi^b_{e'}(\ell)\downarrow$. A contradiction. □

It follows that $\{a, b\}$ is an exact pair for $\{x_n : n < \omega\}$.

□

Definition 11.5. *$\mathcal{I} \subseteq 2^\omega$ is a Turing ideal iff for every finite $\{x_k : k \le N\} \subseteq \mathcal{I}$ and $y \in 2^\omega$, if $y \le_T \bigoplus_{k \le N} x_k$, then $y \in \mathcal{I}$. For $X \subseteq 2^\omega$, the Turing ideal generated X is defined by*

$$\mathcal{I}_X = \left\{ y \in 2^\omega : (\exists\{x_k : k \le N\} \subseteq I) \left(y \le_T \bigoplus_{k \le N} x_k \right) \right\}.$$

Corollary 11.6. *Let $\mathcal{I}$ be a Turing ideal. Then $\mathcal{I}$ has a $\le_T$-least upper bound iff it is finitely generated.*

Definition 11.7 (Minimal pair)**.** *Let $\mathbf{a}, \mathbf{b}, \mathbf{c} \in \mathcal{D}$. We say that $\mathbf{b}, \mathbf{c}$ form a minimal pair above $\mathbf{a}$ iff $\mathbf{a} <_T \mathbf{b}$, $\mathbf{a} <_T \mathbf{c}$ and for every $\mathbf{e} \in \mathcal{D}$,*

$$(\mathbf{e} \le_T \mathbf{b} \text{ and } \mathbf{e} \le_T \mathbf{c}) \implies \mathbf{e} \le_T \mathbf{a}.$$

Exercise 11.8. *Show that for every $\mathbf{a} \in \mathcal{D}$, there exists a minimal pair above $\mathbf{a}$.*

12 Minimal degrees

Definition 12.1 (Sacks forcing)**.** *Let $\mathbb{S}$ be the poset that consists of all perfect trees $p \subseteq {}^{<\omega}2$ ordered by $p \leq_{\mathbb{S}} q$ iff $p \subseteq q$.*

Let $p \in \mathbb{S}$. Recall that σ is a splitting node of p iff $\{\sigma 0, \sigma 1\} \subseteq p$. By induction on $k < \omega$, define the set of **kth level splitting nodes in** p, denoted $\mathsf{splitnode}_k(p)$, as follows.

- $\mathsf{splitnode}_0(p) = \{\sigma\}$ where σ is the $\preceq$-least splitting node of p. σ is called the **stem** of p.
- $\mathsf{splitnode}_{k+1}(p)$ is the set of all splitting nodes $\tau \in p$ such that for some $\sigma \in \mathsf{splitnode}_k(p)$, $\sigma \prec \tau$ and there is no splitting node $\rho \in p$ such that $\sigma \prec \rho \prec \tau$.

It is easy to check that $|\mathsf{splitnode}_k(p)| = 2^k$.

Exercise 12.2 (Fusion)**.** *Suppose $\langle p_n : n < \omega \rangle$ satisfies the following.*

(a) Each $p_n \in \mathbb{S}$ and $p_{n+1} \subseteq p_n$ for all n.

(b) For every $n < \omega$, $\mathsf{splitnode}_n(p_n) = \mathsf{splitnode}_n(p_{n+1})$.

Put $p = \bigcap_{n<\omega} p_n$. Then $p \in \mathbb{S}$.

Definition 12.3 (Recursively pointed perfect trees)**.** *$\mathbb{S}_{rp}$ is the sub-poset of $\mathbb{S}$ that consists of all $p \in \mathbb{S}$ such that $(\forall x \in [p])(p \leq_T x)$. Members of $\mathbb{S}_{rp}$ are called recursively pointed perfect trees.*

Exercise 12.4. *Let $p \in \mathbb{S}_{rp}$. The following hold.*

(a) For every $y \in 2^\omega$, $p \leq_T y$ iff $(\exists x \in [p])(x \equiv_T y)$.

(b) For every $y \in 2^\omega$, if $p \leq_T y$, then there exists $q \in \mathbb{S}_{rp}$ such that $q \subseteq p$ and $q \equiv_T y$.

(c) If $q \in \mathbb{S}$, $q \subseteq p$ and $q \leq_T p$, then $q \in \mathbb{S}_{rp}$ and $q \equiv_T p$.

Definition 12.5. *Let $p \in \mathbb{S}$ and $e < \omega$. We say that p is an e-splitting tree iff for every $x \neq y$ in $[p]$, there exists $k < \omega$ such that $\Phi_e^x(k) \downarrow \neq \Phi_e^y(k) \downarrow$.*

Suppose p is e-splitting, $x \in [p]$ and $\Phi_e^x = y \in 2^\omega$. We claim that $x \leq_T y \oplus p$. To see this, suppose $\sigma = x \restriction n$ has been computed and we want to know whether $x(n)$ is 0 or 1. We can assume that both $\sigma 0, \sigma 1$ are in p otherwise this is easy. As p is e-splitting and $\Phi_e^x = y$, we can perform a successful search for some $\ell < 2$, $\langle \tau_i : i < N \rangle$, $\langle k_i : i < N \rangle$ and $\rho \in p$ such that each real in $[p]$ above $\sigma^\frown \ell$ extends some τ_i, $\sigma^\frown(1-\ell) \preceq \rho$ and $\Phi_e^{\tau_i}(k_i) \downarrow \neq \Phi_e^{\rho}(k_i) \downarrow = y(k_i)$. Then $x(n) = 1 - \ell$.

Definition 12.6. *Let $p \in \mathbb{S}$ and $e < \omega$. We say that p is an e-good tree iff either p is e-splitting or for every $\sigma, \tau \in p$ and $k < \omega$, if $\Phi_e^\sigma(k)$ and $\Phi_e^\tau(k)$ both converge, then they are equal.*

Suppose p is e-good and not e-splitting. Then for every $x \in [p]$, if $\Phi_e^x = y \in 2^\omega$, then $y \leq_T p$ since to compute $y(k)$, we perform a (successful) search for some $\tau \in p$ such that $\Phi_e^\tau(k) \downarrow = \ell$ and output ℓ.

Theorem 12.7 (Spector)**.** *Let $p \in \mathbb{S}_{rp}$.*

(a) For every $e < \omega$, there exists $q \in \mathbb{S}_{rp}$ such that $q \subseteq p$, $q \equiv_T p$ and q is e-good. It follows that for every $x \in [q]$, if $\Phi_e^x = y \in 2^\omega$, then either $y \leq_T q$ or $x \leq_T y \oplus q$.

(b) There exists $r \in \mathbb{S}$ such that $r \subseteq p$ and for every $x \in [r]$ and $y \in 2^\omega$, if $y \leq_T x$, then either $y \leq p$ or $x \leq_T y \oplus p$.

Proof. (a) Call $\sigma \in p$ an ambiguous node if there are there are $k < \omega$ and $\tau_1, \tau_2 \in p$ above σ such that $\Phi_e^{\tau_1}(k) \downarrow \neq \Phi_e^{\tau_2}(k) \downarrow$. We consider two cases.

Case 1. Some $\sigma \in p$ is not ambiguous. Define $q = \{\tau \in p : \tau \preceq \sigma \text{ or } \sigma \preceq \tau\}$. Clearly $q \in \mathbb{S}$ and $q \leq_T p$. So by Exercise 12.4(c), $q \in \mathbb{S}_{rp}$ and $q \equiv_T p$. That q is e-good follows from the fact that σ is not ambiguous.

Case 2. All nodes in p are ambiguous. Inductively construct a sequence $\langle p_n : n < \omega \rangle$ of members of $\mathbb{S}_{rp}$ as follows.

(i) $p_0 = p$.

(ii) Given p_n, define p_{n+1} as follows. Let $\{\sigma_j : j < 2^{n+1}\}$ list $\mathsf{splitnode}_{n+1}(p_n)$. For each $j < 2^n$, search for the least $k < \omega$ and $\tau_{j,0}, \tau_{j,1} \in p$ above σ_j such that $\Phi_e^{\tau_{j,0}}(k) \downarrow \neq \Phi_e^{\tau_{j,1}}(k) \downarrow$ and define

$$p_{n+1} = \{\sigma \in p : (\exists \ell < 2)(\exists j < 2^{n+1})(\sigma \preceq \tau_{j,\ell} \text{ or } \tau_{j,\ell} \preceq \sigma)\}.$$

Put $q = \bigcap_{n<\omega} p_n$ and observe that $q \in \mathbb{S}$ and $q \leq_T p$. So by Exercise 12.4(c), $q \in \mathbb{S}_{rp}$ and $q \equiv_T p$. Note that for every $x \neq y$ in $[q]$, there exists $k < \omega$ such that $\Phi_e^x(k) \downarrow \neq \Phi_e^y(k) \downarrow$. So q is e-splitting and therefore e-good.

(b) Using (a), construct a sequence $\langle p_n : n < \omega \rangle$ of members of $\mathbb{S}_{rp}$ as follows. $p_0 = p$, $p_{n+1} \subseteq p_n$, $p_{n+1} \equiv_T p_n$, $\mathsf{splitnode}_n(p_n) = \mathsf{splitnode}_n(p_{n+1})$ and for each $\sigma \in \mathsf{Split}_{n+1}(p_{n+1})$, $\{\tau \in p_{n+1} : \tau \preceq \sigma \text{ or } \sigma \preceq \tau\}$ is n-good. Define $r = \bigcap_{n<\omega} p_n$. Define $r = \bigcap_{n<\omega} p_n$. Then r is as required. ☐

Corollary 12.8. *Let $a \in 2^\omega$. There exists $r \in \mathbb{S}$ such that for every $x \in [r]$, $a <_T x$ and there is no $y \in 2^\omega$ such that $a <_T y <_T x$. In particular, there is a perfect set $P \subseteq 2^\omega$ such that for each $x \in P$,*

(1) x is not computable and

(2) there is no $y \in 2^\omega$ such that $0 <_T y <_T x$.

Proof. By Exercise 12.2(b) applied to $p = {}^{<\omega}2$, we can choose $q \in \mathbb{S}_{rp}$ such that $q \equiv_T a$. By Theorem 12.7, we can fix $r \in \mathbb{S}$ such that $r \subseteq q$ and for every $x \in [r]$ and $y \leq_T x$, either $y \leq_T q \equiv_T a$ or $x \leq_T y \oplus a$. By throwing away a countable subset of $[r]$, we can assume that no real in $[r]$ is computable from a [Why?]. Fix any $x \in [r]$ and towards a contradiction, suppose there exists $y \in 2^\omega$ such that $a <_T y <_T x$. Then both $y \leq_T a$ and $z \leq_T y \oplus a$ lead to a contradiction. ☐

Corollary 12.9. *For every $\mathbf{a} \in \mathcal{D}$, there exists $\mathbf{b} \in \mathcal{D}$ such that $\mathbf{a} <_T \mathbf{b}$ and there is no $\mathbf{e} \in \mathcal{D}$ such that $\mathbf{a} <_T \mathbf{e} <_T \mathbf{b}$. We say that $\mathbf{b}$ is a* ***minimal cover of*** $\mathbf{a}$.

Corollary 12.10. *There exists $\mathbf{b} \in \mathcal{D}$ such that $\mathbf{0} <_T \mathbf{b}$ and there is no $\mathbf{e} \in \mathcal{D}$ such that $\mathbf{0} <_T \mathbf{e} <_T \mathbf{b}$. We say that $\mathbf{b}$ is a* ***minimal degree****.*

We say that $\mathbf{b}$ is a **strong minimal cover of a** iff $\mathbf{b} >_T \mathbf{a}$ and $(\forall \mathbf{e} \in \mathcal{D})(\mathbf{e} \leq \mathbf{a} \iff \mathbf{e} <_T \mathbf{b})$. The following is a long-standing open problem.

Question 12.11 (Yates)**.** *Suppose $\mathbf{a}$ is a minimal degree. Must there exist $\mathbf{b}$ such that $\mathbf{b}$ is a strong minimal cover of $\mathbf{a}$?*

Theorem 12.12 (Sacks)**.** *Let $A \subseteq 2^\omega$ be countable. Then there exists $p \in \mathbb{S}$ such that for every $y \in [p]$, the following hold.*

(i) y computes every real in $\mathcal{I}_A$.

(ii) If $z \leq_T y$, then there exists $x \in \mathcal{I}_A$ such that either $z \leq_T x$ or $y \leq_T x \oplus z$.

(iii) If $\mathcal{I}_A$ is not finitely generated, then y is a $\leq_T$-minimal upper bound of $\mathcal{I}_A$.

Proof. Fix a $\leq_T$-increasing sequence $\langle a_k : k < \omega \rangle$ such that $a_0 \equiv_T 0$, each $a_k \in \mathcal{I}_A$ and $(\forall y \in \mathcal{I}_A)(\exists k < \omega)(y \leq_T a_k)$. We can further assume that if $\mathcal{I}_A$ is not finitely generated, then $a_n <_T a_{n+1}$ for every n.

Inductively construct a sequence $\langle p_n : n < \omega \rangle$ of members of $\mathbb{S}_{rp}$ satisfying the following.

(1) $p_0 = {}^{<\omega}2$.

(2) $p_n \in \mathbb{S}_{rp}$ and $p_{n+1} \subseteq p_n$.

(3) $\mathsf{splitnode}_n(p_n) = \mathsf{splitnode}_n(p_{n+1})$.

(4) For every $\sigma \in \mathsf{Split}_{n+1}(p_{n+1})$, $\{\tau \in p_{n+1} : \sigma \preceq \tau \text{ or } \tau \preceq \sigma\}$ is n-good.

(5) $p_{n+1} \equiv_T a_{n+1}$.

To obtain p_{n+1} from p_n, we use Theorem 12.7(a) and Exercise 12.4(b). Let $p' \in \mathbb{S}$ be the intersection of $\{p_n : n < \omega\}$. Choose $p \in \mathbb{S}$ such that $p \subseteq p'$ and no member of $[p]$ is computable from any member of $\mathcal{I}_A$.

Let us check that p is as required. Fix $y \in [p]$. Since $\{a_n : n < \omega\}$ is $\leq_T$-cofinal in $\mathcal{I}_A$, it is clear than y computes every real in $\mathcal{I}_A$. Thus (i) holds. Next, assume $z \leq_T y$ and fix $e < \omega$ such that $\Phi_e^y = z$. Fix $n > e + 1$. Then by Clauses (4) and (5), either $z \leq_T a_n$ or $y \leq_T z \oplus a_n$. So (ii) holds. Finally assume $\mathcal{I}_A$ is not finitely generated. Then $a_n <_T a_{n+1}$ for every n. Towards a contradiction, fix some $z <_T y$ such that $(\forall n)(a_n \leq_T z)$. Using (ii), fix $N < \omega$ such that either $z \leq_T a_N$ (impossible since $a_N <_T a_{N+1} \leq_T z$) or $y \leq_T z \oplus a_N$. As $a_N \leq_T z$, the latter implies $y \leq_T z <_T y$ which is a contradiction. $\square$

Corollary 12.13. *Let $\langle a_n : n < \omega\rangle$ be $<_T$-strictly increasing sequence of reals. Then there exists a perfect set $P \subseteq 2^\omega$ such that each member of P is a $\leq_T$-minimal upper bound of $\{a_n : n < \omega\}$.*

13 Incomparable c.e. sets

Definition 13.1. *We write $\Phi_e^A(n)[s]\downarrow$ iff the oracle use of the computation $\Phi_e^A(n)$ is contained in $A \upharpoonright s$ and the number of steps before the computation $\Phi_e^A(n)$ halts is less than s.*

Theorem 13.2 (Friedberg-Muchnik)**.** *There exist c.e. sets A, B such that $A \not\leq_T B$ and $B \not\leq_T A$.*

Proof. Our *requirements* are $R_{2e} : (\exists n)(\Phi_e^B(n) \neq A(n))$ (this means that either $\Phi_e^B(n)\uparrow$ or $\Phi_e^B(n)\downarrow\neq A(n)$) and $R_{2e+1} : (\exists n)(\Phi_e^A(n) \neq B(n))$. To ensure that A, B are c.e., we will construct a uniformly computable *$\subseteq$-increasing* sequence $\langle A_s, B_s : s < \omega\rangle$ of finite subsets of ω and set $A = \bigcup_s A_s$ and $B = \bigcup_s B_s$.

What is our strategy to satisfy the requirements? Say we are at stage $s+1$ and we want to satisfy R_{2e}. Since we can only *add* new numbers to A_s, it is reasonable to try to see if for some $n_e < s$, $\Phi_e^{B_s}(n_e)[s]\downarrow= 0$ in which case we define $A_{s+1} = A_s \cup \{n_e\}$, $B_{s+1} = B_s$ and say that R_{2e} has *acted* at stage $s+1$. Note that the requirement R_{2e} will continue to remain satisfied as long as we do not add any $n < s$ to B_{s+1} at a later stage.

Injury. Suppose a requirement R_i has acted at stage s and at some later stage $t > s$ another requirement R_j acts. It may happen that the action of R_j changes the oracle use of the computation that satisfied R_i. In this case we say that R_i *has been injured* and it might have to act again at a later stage.

Priority. To minimize injuries, we will require that for each requirement, there is a stage beyond which it never gets injured. This is easily arranged by prioritizing the requirements as follows. Suppose a requirement R_i has acted at stage s and at some later stage $t > s$ another requirement R_j acts where $j > i$. We then require that R_j *does not add* any number into A_t or B_t below the oracle use of the computation that satisfies R_i. This means that R_i can only be injured by R_j when $j < i$. In other words R_i has *higher priority* than R_j when $i < j$.

We are going to define a uniformly computable sequence $\langle A_s, B_s, \overline{n}_s, f_s : s < \omega\rangle$ where

(a) A_s, B_s are finite and $\subseteq$-increasing with s.

(b) $\overline{n}_s = \langle n_{i,s} : i < \omega\rangle$ where each $n_{i,s} < \omega$. Think of $n_{i,s}$ as a witness at which R_i is supposed to be satisfied. We will ensure below that for every $i < j < \omega$, $\{n_{i,s} : s < \omega\} \cap \{n_{j,s} : s < \omega\} = \emptyset$.

(c) $f_s : \omega \to \{0, 1\}$. We interpret $f_s(i) = 1$ as "R_i needs attention at stage s" and $f_s(i) = 0$ as "R_i does not need attention at stage s"

Stage $s = 0$**.** Define $A_0 = B_0 = \emptyset$, $x_{i,0} = \langle i, 0\rangle$ and $f_0(i) = 1$ for all $i < \omega$.

Stage $s + 1$**.** We first ask: Is there an $i < s$ such that the following hold.

(A) $f_s(i) = 1$ (R_i needs attention).

(B) If $i = 2e$, then $\Phi_e^{B_s}(n_{s,i})[s]\downarrow= 0$ and if $i = 2e + 1$, then $\Phi_e^{A_s}(n_{s,i})[s]\downarrow= 0$.

If there is no such $i < s$, then define $A_{s+1} = A_s$, $B_{s+1} = B_s$, $\overline{n}_{s+1} = \overline{n}_s$ and $f_{s+1} = f_s$. Otherwise, fix the least such i and do the following. We will say that R_i **has acted at stage** $s+1$.

(i) If $i = 2e$, then define $A_{s+1} = A_s \cup \{n_{s,i}\}$ and $B_{s+1} = B_s$. If $i = 2e+1$, then define $B_{s+1} = B_s \cup \{n_{s,i}\}$ and $A_{s+1} = A_s$.

(ii) Define $n_{s+1,i} = n_{s,i}$ and $f_{s+1}(i) = 0$ (R_i has acted and does not need attention at stage $s+1$).

(iii) For each $j > i$, define $f_{s+1}(j) = 0$ (R_j is injured and will require attention) and $n_{s+1,j} = \langle n_s, s+1\rangle > s+1$ (so R_j cannot injure R_i).

(iv) For each $j < i$, define $n_{s+1,j} = n_{s,j}$ and $f_{s+1}(j) = f_s(j)$ (R_j's status has not changed as it has higher priority than R_i).

Put $A = \bigcup_s A_s$ and $B = \bigcup_s B_s$ and note that both A, B are c.e. Let us check that A, B satisfy all the requirements. Note that if R_i acts at stage $s+1$, then $f_s(i) = 1$ and $f_{s+1}(i) = 0$. So if $\lim_{s\to\infty} f_s i$ converges, then R_i acts only finitely many times.

Claim 13.3. *For each i, there exist $n_i < \omega$ and $k_i \in \{0,1\}$ such that $\lim_{s\to\infty} n_{i,s} = n_i$ and $\lim_{s\to\infty} f_s(i) = k_i$. Also, A, B satisfy R_i for every $i < \omega$.*

Proof. By induction on i. Suppose $i = 0$. Recall that $n_{0,0} = \langle 0,0\rangle = 0$ and $f_0(0) = 1$. We consider two cases. Either R_0 acts at some stage or it never acts.

Say R_0 acts for the first time at stage $s+1$. Then Clause (iv) above guarantees that R_0 will never act at any stage $t > s+1$. Hence for every $t \geq s+1$, $f_t(0) = 0 = k_0$ and $n_{0,t} = n_{0,0} = n_0$. Also observe that $\Phi_0^B(n_0)\downarrow = 0$ and $A(n_0) = 1$. So R_0 is satisfied.

Now assume that R_0 never acts. Then by Clause (iv), we must have $f_s(0) = f_0(0) = 1 = k_0$ and $n_{0,s} = n_{0,0} = n_0$ for all s. Since $n_0 \notin \{n_{i+1,s} : i, s < \omega\}$, we must have $n_0 \notin A$. It follows that R_0 is satisfied as witnessed by n_0 since either $\Phi_0^B(n_0)\uparrow$ or $\Phi_0^B(n_0)\downarrow\neq 0$ and $n_0 \notin A$.

Now assume that the Claim holds for all $i \leq j$. Choose a stage s large enough so that $f_s(i) = f_t(i)$ and $n_{i,s} = n_{t,i}$ for all $t > s$ and $j \leq i$. Then for every $i \leq j$ R_i does not act at any stage $\geq s$. We consider two cases. R_j acts at some stage $t \geq s$ or R_j does not at any stage $t \geq s$. Now argue as we did for R_0. □

Recall that a c.e. degree is the Turing degree of a c.e. set.

Corollary 13.4. *There exists a c.e. degree* $\mathbf{a}$ *such that* $\mathbf{0} <_T \mathbf{a} < \mathbf{0}'$.

Exercise 13.5. *Show that there is an infinite Turing independent family of of c.e. sets.*

14 Generic reals

Let $\mathbb{P} = {}^{<\omega}2$ be Cohen forcing and $D \subseteq \mathbb{P}$. Recall that $D \subseteq {}^{<\omega}2$ is dense iff every $\sigma \in {}^{<\omega}2$, there exists $\tau \in D$ such that $\sigma \preceq \tau$.

Exercise 14.1. *Let $D \subseteq {}^{<\omega}2$ be dense. Show that $\bigcup_{\sigma\in D}[\sigma]$ is an open dense subset of 2^ω.*

Definition 14.2 (Weakly 1-generic)**.** *$x \in 2^\omega$ is weakly 1-generic iff for every dense c.e. $D \subseteq {}^{<\omega}2$, there exists $\sigma \in D$ such that $\sigma \preceq x$.*

Definition 14.3. *$U \subseteq 2^\omega$ is a c.e. open set iff there is a c.e. $A \subseteq {}^{<\omega}2$ such that*

$$U = \bigcup_{\sigma\in A}[\sigma].$$

Exercise 14.4. *Show that $x \in 2^\omega$ is weakly 1-generic iff for every open dense c.e. set $U \subseteq 2^\omega$, $x \in U$.*

Let $\mathcal{F}$ be the family of all c.e. open dense sets $U \subseteq 2^\omega$. Then $\mathcal{F}$ is countable. Hence $\bigcap_{U\in\mathcal{F}} U$ is comeager in 2^ω. So we have the following.

Lemma 14.5. *$\{x \in 2^\omega : x$ is not weakly 1-generic$\}$ is meager.*

Suppose $A \subseteq \omega$ is infinite. The **principal function of** A, (denoted p_A) is defined by: $p_A(n)$ is the nth member of A. So $A = \{p_A(0) < p_A(1) < \cdots\}$.

Definition 14.6 (Hyperimmune set)**.** *$A \subseteq \omega$ is hyperimmune iff A is infinite and for every computable $f : \omega \to \omega$, $(\exists^\infty n)(f(n) < p_A(n))$.*

Note that $A \subseteq B \implies (\forall n)(p_A(n) \leq p_B(n))$. It follows that every infinite subset of a hyperimmune set is also hyperimmune.

Exercise 14.7. *Show that every hyperimmune set is immune.*

Suppose $f, g : \omega \to \omega$. We say that f **majorizes** g iff $(\forall n)(f(n) \geq g(n))$. We say that f **dominates** g iff $(\exists N)(\forall n \geq N)(f(n) \geq g(n))$.

Exercise 14.8. *Let $A \subseteq \omega$ be infinite. Show that the following are equivalent.*

(a) A is hyperimmune.

(b) p_A is not dominated by any computable $f : \omega \to \omega$.

(c) p_A is not majorized by any strictly increasing computable $f : \omega \to \omega$.

Lemma 14.9. *Every weakly 1-generic real is (the characteristic function of) a hyperimmune set.*

Proof. Let $x \in 2^\omega$ be weakly 1-generic. Let $f : \omega \to \omega$ be a strictly increasing computable function. By the previous exercise, it suffices to find some ℓ such that $f(\ell) < p_x(\ell)$.

Define $A \subseteq {}^{<\omega}2$ as follows: $\sigma \in A$ iff $|\sigma| = k$, $\sigma(k) = 1$ and $f(\ell) < k$ where $\ell = |\{j < k : \sigma(j) = 1\}|$. It is easy to check that A is computable and dense in ${}^{<\omega}2$. As x is weakly 1-generic, there exists $\sigma \in A$ such that $\sigma \preceq x$. Let $\ell = |\{j < |\sigma| - 1 : \sigma(j) = 1\}|$. Then $p_x(\ell) > f(\ell)$. ☐

Since a c.e. set cannot be immune, we get the following.

Corollary 14.10. *No c.e. set is weakly 1-generic.*

Definition 14.11 (1-generic)**.** *$x \in 2^\omega$ is 1-**generic** iff for every c.e. $S \subseteq {}^{<\omega}2$, either $(\exists \sigma \in S)(\sigma \preceq x)$ or $(\exists n)(\forall \sigma)(s \restriction n \preceq \sigma \implies \sigma \notin S)$.*

For $S \subseteq {}^{<\omega}2$, define

$$S_\star = S \cup \{\sigma \in {}^{<\omega}2 : (\forall \tau \in {}^{<\omega}2)(\sigma \preceq \tau \implies \tau \notin S)\}.$$

Exercise 14.12. *Show that for every $S \subseteq {}^{<\omega}2$, $S_\star$ is dense in ${}^{<\omega}2$ and S is dense in ${}^{<\omega}2$ iff $S = S_\star$. Furthermore, $x \in 2^\omega$ is 1-generic iff for every c.e. $S \subseteq {}^{<\omega}2$,*

$$x \in \bigcup_{\sigma \in S_\star} [\sigma].$$

It follows that $\{x \in 2^\omega : x$ is not 1-generic$\}$ is meager.

Exercise 14.13. *Show that every 1-generic real is weakly 1-generic.*

Definition 14.14 (Generalized low)**.** *A is GL_1 iff $A \oplus 0' \equiv_T A'$.*

Theorem 14.15. *Every 1-generic is GL_1.*

Proof. Clearly, $A \oplus 0' \leq_T A'$. So it suffices to show that $A' \leq_T A \oplus 0'$. For each $e < \omega$, define $S_e = \{\sigma \in {}^{<\omega}2 : \Phi_e^\sigma(e)\downarrow\}$. Then S_e is a c.e. set. As A is 1-generic, there exists $\sigma \preceq A$ such that either $\sigma \in S_e$ or no extension of σ is in S_e.

Put $W = \{(\sigma, e) : (\exists \tau)(\sigma \preceq \tau$ and $\Phi_e^\tau(e)\downarrow)\}$. Then W is c.e. and therefore $W \leq_T 0'$.

Now consider the oracle program P (using $A \oplus 0'$ as oracle) that on input e, searches for some $\sigma \preceq A$ such that either $\Phi_e^\sigma(e)\downarrow$ or $(\sigma, e) \notin W$. In the former case, P outputs 1 and in the latter case, it outputs 0. It is easy to see that P computes A'. ☐

Definition 14.16. *$A \subseteq \omega$ is low iff $A' \equiv_T 0'$.*

Lemma 14.17. *There exists a 1-generic $x \in 2^\omega$ such that $x \le_T 0'$. Any such x is low.*

Proof. First check that $W \le_T 0'$ where

$$W = \{(n, \sigma) : (\varphi_n(\sigma)\downarrow) \text{ or } (\forall \tau \succeq \sigma)(\varphi_n(\tau)\uparrow)\}.$$

Construct a $0'$-computable sequence $\langle \sigma_n : n < \omega \rangle$ as follows.

(1) Define $\sigma_0 = \emptyset$.

(2) Suppose σ_n has been defined. Let $S = \{\sigma \in {}^{<\omega}2 : \varphi_n(\sigma)\downarrow\}$ be the nth c.e. subset of ${}^{<\omega}2$. We would like to choose some $\sigma_{n+1} \in S_\star$ such that $\sigma_n \prec \sigma_{n+1}$. Since $S_\star$ is dense in ${}^{<\omega}2$ and $W \le_T 0'$, using $0'$ as the oracle, we can search for the least $\sigma \in {}^{<\omega}2$ (under some computable enumeration of ${}^{<\omega}2$) such that $\sigma_n \prec \sigma$ and $(n, \sigma) \in W$ and define $\sigma_{n+1} = \sigma$.

It is easy to check that $x = \bigcup_{n<\omega} \sigma_n$ is 1-generic. □

Lemma 14.18. *Let $A \subseteq \omega$ be infinite. Then A is hyperimmune iff for some 1-generic $B \subseteq \omega$, $A \subseteq B$.*

Proof. Since every infinite subset of a hyperimmune set is also hyperimmune and every 1-generic is hyperimmune (Lemma 14.9 + Exercise 14.13), the right to left implication is clear.

For the converse, fix a hyperimmune $A \subseteq \omega$. Construct $\langle \sigma_n : n < \omega \rangle$ as follows.

(1) $\sigma_0 = \emptyset$.

(ii) Suppose $\sigma_n \in {}^{<\omega}2$ has been defined such that $(\forall i < |\sigma_n|)(i \in A \implies \sigma_n(i) = 1)$. Let $S = \{\sigma \in {}^{<\omega}2 : \varphi_n(\sigma)\downarrow\}$ be the nth c.e. subset of ${}^{<\omega}2$. It suffices to find $\sigma_{n+1} \in S_\star$ such that $\sigma_n \preceq \sigma_{n+1}$ and $(\forall i < |\sigma_{n+1}|)(i \in A \implies \sigma_{n+1}(i) = 1)$. We consider two cases.

Case A: There exists $k \ge 1$ such that $(\forall \tau \succeq \sigma_n^\frown 1^k)(\tau \notin S)$. In this case, define $\sigma_{n+1} = \sigma_n^\frown 1^k$.

Case B: For every $k \ge 1$, $(\exists \tau \succeq \sigma_n^\frown 1^k)(\tau \in S)$. In this case, we can recursively construct a strictly increasing computable $f : \omega \to \omega$ and a computable sequence $\langle \tau_k : k < \omega \rangle$ such that for each k, $\tau_k \in S$, $\sigma_n^\frown 1^{f(k)} \preceq \tau_k$ and $|\tau_k| < f(k+1)$. As A is hyperimmune, we can find some k such that the interval $[f(k), f(k+1))$ is disjoint from A. So we can take $\sigma_{n+1} = \tau_k$.

It is cleat that $x = \bigcup_{n<\omega} \sigma_n$ is 1-generic and $A \subseteq B$ where $1_B = x$. □

Definition 14.19. *Let $x, y \in 2^\omega$. We say that x is 1-generic relative to y iff for every $S \subseteq {}^{<\omega}2$, if S is c.e. in x, then there exists $\sigma \preceq x$ such that either $\sigma \in S$ or no extension of σ is in S.*

Note that if x is 1-generic relative to y, then x is 1-generic. Also, for every y,

$$\{x \in 2^\omega : x \text{ is not 1-generic relative to } y\} \text{ is meager.}$$

Lemma 14.20. *Suppose $x \in 2^\omega$ is 1-generic. Let $x = x_0 \oplus x_1$ (here, $x_0(n) = x(2n)$ and $x_1(n) = x(2n+1)$ for all n). Then x_0 is 1-generic relative to x_1 and x_1 is 1-generic relative to x_0.*

Proof. Let $S \subseteq {}^{<\omega}2$ be c.e. in x_1. We need to find $\sigma_0 \preceq x_0$ such that either $\sigma_0 \in S$ or no extension of σ_0 is in S.

Since S is c.e. in y, we can fix $e < \omega$ such that for each $\sigma \in {}^{<\omega}2$,

$$\sigma \in S \iff \Phi_e^{x_1}(\sigma)\downarrow \iff (\exists \rho \preceq x_1)(\Phi_e^{\rho}(\sigma)\downarrow).$$

Consider the set W that consists of all $\sigma \in {}^{<\omega}2$ such that letting $\sigma = \sigma_0 \oplus \sigma_1$, we have $(\exists \tau \preceq \sigma_0)\Phi_e^{\sigma_1}(\tau)\downarrow$. As W is c.e. and x is 1-generic, there must exist $\sigma \preceq x$ such that either $\sigma \in W$ or no extension of σ is in W. Fix such $\sigma \preceq x$ and put $\sigma = \sigma_0 \oplus \sigma_1$.

First assume $\sigma \in W$. Then

$$\sigma \in W \implies (\exists \tau \preceq \sigma_0)(\Phi_e^{\sigma_1}(\tau)\downarrow) \implies (\exists \tau \preceq \sigma_0)(\Phi_e^{x_1}(\tau)\downarrow) \implies \tau \in S.$$

Next assume that no extension of σ is in W. Then for every $\tau \succeq \sigma_0$, $\Phi_e^{x_1}(\tau)\uparrow$. So $\tau \notin S$. It follows that no extension on σ_0 is in S and we are done. □

Exercise 14.21. *Suppose $x \in 2^\omega$ is 1-generic. Let $x = x_0 \oplus x_1$ (here, $x_0(n) = x(2n)$ and $x_1(n) = x(2n+1)$ for all n). Then x_0, x_1 form a minimal pair (Definition 11.7).*

15 Arithmetical hierarchy

Definition 15.1 (Arithmetical hierarchy). *By induction on $n < \omega$, for each $k \geq 1$, we define the classes $\Sigma^0_n(\omega^k)$, $\Pi^0_n(\omega^k)$ and $\Delta^0_n(\omega^k)$ as follows.*

(1) $A \in \Sigma^0_0(\omega^k)$ iff $A \in \Pi^0_0(\omega^k)$ iff $A \subseteq \omega^k$ is computable.

(2) $A \in \Sigma^0_{n+1}(\omega^k)$ iff there exists $R \in \Pi^0_n(\omega^{k+1})$ such that for every $(x_1, \cdots, x_k) \in \omega^k$,

$$(x_1, \cdots, x_k) \in A \iff (\exists y)((y, x_1, \cdots, x_k) \in R).$$

(3) $A \in \Pi^0_{n+1}(\omega^k)$ iff there exists $R \in \Sigma^0_n(\omega^{k+1})$ such that for every $(x_1, \cdots, x_k) \in \omega^k$,

$$(x_1, \cdots, x_k) \in A \iff (\forall y)((y, x_1, \cdots, x_k) \in R).$$

(4) $A \in \Delta^0_n(\omega^k)$ iff $A \in \Sigma^0_n(\omega^k)$ and $A \in \Pi^0_n(\omega^k)$.

Relativizing the arithmetical hierarchy to an oracle X yields the following notions.

Definition 15.2. *Let $X \subseteq \omega$. By induction on $n < \omega$, for each $k \geq 1$, we define the classes $\Sigma^X_n(\omega^k)$, $\Pi^X_n(\omega^k)$ and $\Delta^X_n(\omega^k)$ as follows.*

(1) $A \in \Sigma^X_0(\omega^k)$ iff $A \in \Pi^X_0(\omega^k)$ iff $A \subseteq \omega^k$ is computable in X.

(2) $A \in \Sigma^X_{n+1}(\omega^k)$ iff there exists $R \in \Pi^X_n(\omega^{k+1})$ such that for every $(x_1, \cdots, x_k) \in \omega^k$,

$$(x_1, \cdots, x_k) \in A \iff (\exists y)((y, x_1, \cdots, x_k) \in R).$$

(3) $A \in \Pi^X_{n+1}(\omega^k)$ iff there exists $R \in \Sigma^X_n(\omega^{k+1})$ such that for every $(x_1, \cdots, x_k) \in \omega^k$,

$$(x_1, \cdots, x_k) \in A \iff (\forall y)((y, x_1, \cdots, x_k) \in R).$$

(4) $A \in \Delta^X_n(\omega^k)$ iff $A \in \Sigma^X_n(\omega^k)$ and $A \in \Pi^X_n(\omega^k)$.

Exercise 15.3. *Show the following.*

(i) The classes Σ^X_n, Π^X_n and Δ^X_n are increasing with n.

(ii) The classes Σ^X_n, Π^X_n and Δ^X_n are closed under finite intersections and unions.

(iii) The classes Σ^X_n, Π^X_n and Δ^X_n are downward closed under $\leq_m$. So for example, if $A \in \Sigma^X_n$ and $B \leq_m A$, then $B \in \Sigma^X_n$.

(iv) The classes Σ^X_n, Π^X_n are not downward closed under $\leq_T$. For example, $0' \in \Sigma^0_1$ and $\omega \setminus 0' \leq_T 0'$ but $\omega \setminus 0' \notin \Sigma^0_1$.

Exercise 15.4. *By induction on n, prove the following.*

(a) $A \in \Sigma^X_n(\omega^k)$ iff $(\omega^k \setminus A) \in \Pi^X_n(\omega^k)$.

(b) (Closure under projection) If $A \in \Sigma^X_n(\omega^{k+1})$ and $B \subseteq \omega^k$ is defined by

$$(x_1 \cdots, x_k) \in B \iff (\exists y)((y, x_1, \cdots, x_k) \in A),$$

then $B \in \Sigma^X_n(\omega^k)$.

(c) If $A \in \Pi^X_n(\omega^{k+1})$ and $B \subseteq \omega^k$ is defined by

$$(x_1 \cdots, x_k) \in B \iff (\forall y)((y, x_1, \cdots, x_k) \in A),$$

then $B \in \Pi^X_n(\omega^k)$.

Let $\mathcal{N} = (\omega, 0, S, +, \cdot)$ be the standard model of PA (Peano arithmetic). Recall that $A \subset \omega^k$ is first order definable in $\mathcal{N}$ iff there exists an $\mathcal{L}_{PA}$- formula $\phi(x_1, \cdots, x_k)$ such that for every $(n_1, \cdots, n_k) \in \omega^k$,

$$(n_1, \cdots, n_k) \in A \iff \mathcal{N} \models \phi(n_1, \cdots, n_k).$$

Theorem 15.5. *$A \subseteq \omega^k$ is definable in $\mathcal{N}$ iff for some $n < \omega$, $A \in \Sigma^0_n$.*

When $k = 1$, we will sometimes drop ω^k and write Σ^X_n instead of $\Sigma^X_n(\omega)$ etc. The next lemma can be proved exactly like the results in Section 5

Lemma 15.6. *$A \in \Sigma^X_1$ iff A is c.e. in X $A \leq_1 X'$ iff $A \leq_m X'$.*

Theorem 15.7 (Post)**.** *The following are equivalent for every $n \geq 1$ and $A \subseteq \omega^k$.*

(1) $A \in \Sigma^X_n$.

(2) A is c.e. in X^{n-1}. Here, $X^0 = X$ and X^n is the nth Turing jump of X.

(3) $A \leq_1 X^n$.

(4) $A \leq_m X^n$.

Proof. The equivalence of (2), (3), (4) and $A \in \Sigma^{X^{n-1}}_1$ follows from Lemma 15.6. Now use induction on n to check that $A \in \Sigma^X_n$ iff $A \in \Sigma^{X^{n-1}}_1$. □

Corollary 15.8. *$A \in \Delta^X_n$ iff $A \leq_T X^{n-1}$.*

Proof. $A \in \Delta^X_n$ iff $A \in \Sigma^X_n \cap \Pi^X_n$ iff (Post theorem) $A \leq_1 X^{n-1}$ and $\omega \setminus A \leq_1 X^{n-1}$ iff $A \leq_T X^{n_1}$. □

Corollary 15.9. *For every $n \geq 1$, $0^n \in \Sigma^0_n \setminus \Pi^0_n$.*

Definition 15.10. *$A \subseteq$ is limit computable iff there exists a uniformly computable sequence $\langle A_n : n < \omega \rangle$ of subsets of ω such that $A = \lim_n A_n$.*

Lemma 15.11 (Shoenfield Limit lemma)**.** *A is limit computable iff $A \leq_T 0'$ iff $A \in \Delta^0_2$.*

Proof. That $A \leq_T 0'$ iff $A \in \Delta^0_2$ follows from Corollary 15.8.

Suppose A is limit computable and fix a uniformly computable sequence $\langle A_n : n < \omega \rangle$ such that $A = \lim_n A_n$. Then $x \in A \iff (\exists N)(\forall n)(n \geq N \implies x \in A_n)$. This shows that $A \in \Sigma^0_2$. Similarly, $x \notin A \iff (\exists N)(\forall n)(n \geq N \implies x \notin A_s)$ and so $A \in \Pi^0_2$. Thus $A \in \Delta^0_2$.

Next suppose $A \leq_T K = 0'$ and fix $e < \omega$ such that $\Phi^K_e = 1_A$. Define $K_s = \{n < s : \varphi_e(e)[s] \downarrow\}$. Then $\langle K_s : s < \omega \rangle$ is a uniformly computable sequence of finite sets with union K. Define $A_s = \{n < s : \Phi^{K_s}_e(n)[s] \downarrow= 1\}$. Now check that $A = \lim_s A_s$. □

Exercise 15.12 (Relativized limit lemma)**.** *Call $A \subseteq \omega$ limit computable in X iff there exists a uniformly X-computable sequence $\langle A_s : s < \omega \rangle$ such that $A(x) = \lim_s A_s(x)$. Show that A is limit computable in X iff $A \leq_T X'$.*

Definition 15.13. *$A \subseteq \omega$ is a Σ^0_n-complete set iff $A \in \Sigma^0_n$ and for every $B \in \Sigma^0_n$, $B \leq_1 A$ (equivalently, $0^n \leq_1 A$).*

$A \subseteq \omega$ is a Π^0_n-complete set iff $A \in \Pi^0_n$ and for every $B \in \Pi^0_n$, $B \leq_1 A$ (equivalently, $\omega \setminus 0^n \leq_1 A$).

The next corollary implies that all Σ^0_1-complete sets are computably isomorphic to $0'$.

Corollary 15.14. *0^n is a Σ^0_n-complete set and for every $A \in \Sigma^0_n$, A is Σ^0_n-complete iff $A \equiv_1 0^n$.*

Proof. A is Σ^0_n complete iff $A \in \Sigma^0_n$ and $0^n \leq_1 A$ iff $A \leq_1 0^n$ and $0^n \leq_1 A$ iff $A \equiv_1 0^n$. □

Are there noncomputable c.e. sets that are not Σ^0_1-complete?

Exercise 15.15. *Let A be a simple set. Show that A is not Σ^0_1-complete.*

Lemma 15.16. *For every $A \subseteq \omega$, $A \equiv_m 0'$ iff $A \equiv_1 0'$ iff A is Σ^0_n-complete.*

Proof. The only new implication is $0' \leq_m A \implies 0' \leq_1 A$ which is HW 7. $\square$

Some examples of complete sets follow.

Lemma 15.17. *$\mathsf{Tot} = \{e : W_e = \omega\}$ is Π^0_2-complete.*

Proof. $\mathsf{Tot} \in \Pi^0_2$ since $e \in \mathsf{Tot}$ iff $(\forall n)(\exists s)(\varphi_e(n)[s]\downarrow)$. Suppose $A \in \Pi^0_2$. Fix a computable $R \subseteq \omega^3$ such that $x \in A \iff (\forall y)(\exists z)(R(x,y,z))$. Define $\theta(x,y) = 1$ if $(\exists z)(R(x,y,z))$ and undefined otherwise. Then θ is partial computable so by s^n_m-theorem, there is an injective computable $h : \omega \to \omega$ such that $\theta(x,y) = \varphi_{h(x)}(y)$. Now check that h is a $\leq_1$-reduction from A to Tot. $\square$

Lemma 15.18. *$\mathsf{Fin} = \{e : W_e$ is finite$\}$ is Σ^0_2-complete.*

Proof. $\mathsf{Fin} \in \Sigma^0_2$ as

$$|W_e| < \omega \iff (\exists N)(\forall x \geq N)(\forall s)(\varphi_e(x)[s]\uparrow).$$

Suppose $A \in \Sigma^0_2$. Fix a computable $R \subseteq \omega^3$ such that $x \in A \iff (\exists y)(\forall z)(R(x,y,z))$. Define $\theta(x,y) = 1$ if $(\forall y \leq z)(\exists z)(\neg R(x,y,z))$ and undefined otherwise. Then θ is partial computable so by s^n_m-theorem, there is an injective computable $h : \omega \to \omega$ such that $\theta(x,y) = \varphi_{h(x)}(y)$. Now check that h is a $\leq_1$-reduction from A to Fin. $\square$

Lemma 15.19. *$\mathsf{Cof} = \{e : W_e$ is cofinite$\}$ is Σ^0_3-complete.*

Proof. $\mathsf{Cof} \in \Sigma^0_3$ because

$$W_e \text{ is cofinite} \iff (\exists N)(\forall x \geq N)(\exists s)(\varphi_e(x)[s]\downarrow).$$

Next suppose $A \in \Sigma^0_3$. Since Fin is Σ^0_2-complete, we can find an injective computable $h : \omega^2 \to \omega$ such that $x \in A \iff (\exists x)(W_{h(x,y)}$ is infinite). For each $x < \omega$, define a c.e. set C^x uniformly in x as follows. $C^x = \bigcup_s C^x_s$ will be the union of a uniformly computable sequence $\langle C^x_s : s < \omega \rangle$ defined as follows.

(1) $C^x_0 = \emptyset$.

(2) Let $\omega \setminus C^x_s = \{n^s_0 < n^s_1 < \cdots\}$. For each $y < s$, if $W_{h(x,y),s+1} \setminus W_{h(x,y),s} \neq \emptyset$, then add n^s_y to C^x_s. So

$$C^x_{s+1} = C^x_s \cup \{n^s_y : y < s \text{ and } W_{h(x,y),s+1} \setminus W_{h(x,y),s} \neq \emptyset\}.$$

By padding, we can fix a computable injective $f : \omega \to \omega$ such that $C^x = W_{f(x)}$. Now check that

$$x \in A \iff (\exists y)(W_{h(x,y)} \text{ is infinite}) \iff W_{f(x)} \text{ is cofinite}.$$

The above construction of C^x is an example of a "**movable marker construction**".

$\square$

Exercise 15.20. *$\mathsf{Rec} = \{e : W_e$ is computable$\}$ is Σ^0_3-complete.*

Definition 15.21. *$A \subseteq \omega$ is high iff $0'' \leq_T A'$. $\mathbf{a} \in \mathcal{D}$ is high iff it is the Turing degree of a high set.*

Lemma 15.22. *$A \subseteq \omega$ is high iff $0'' \in \Delta^A_2$ iff $\mathsf{Tot} \leq_T A'$.*

Proof. Use Corollary 15.8 and the fact that Tot is Π^0_2-complete and therefore $\mathsf{Tot} \equiv_T 0''$. $\square$

Theorem 15.23 (Martin). *$A \subseteq \omega$ is high iff there exists $d : \omega \to \omega$ such that $d \leq_T A$ and d dominates every computable function.*

Proof. Suppose A is high. Then by Lemma 15.22, $\mathsf{Tot} \leq_T A'$. By the relativized limit lemma, there is an A-computable sequence $\langle h_s(x) : s < \omega \rangle$ where each $h_s : \omega \to 2$ and $\mathsf{Tot}(x) = \lim_s h_s(x)$. Define $d : \omega \to \omega$ as follows.

For each $x \leq s < \omega$, define $t(x,s)$ to be the least $t > s$ such that either $h_t(x) = 0$ or $(\forall n \leq s)(\varphi_x(n)[t]\downarrow)$. To see that $t(x,s)$ is well-defined, consider the cases $x \in \mathsf{Tot}$ and $x \notin \mathsf{Tot}$. Also note that $t \leq_T A$.

Next define $d(s) = \max\{t(x,s) : x \leq s\}$. Then $d \leq_T A$. Let us check that d dominates every computable function. Suppose φ_e is total. Then $e \in \mathsf{Tot}$. Since $\mathsf{Tot}(e) = \lim_s h_s(e)$, we can fix $s_\star > e$ such that $(\forall t \geq s_\star)(h_t(e) = 1)$. This implies that if $s \geq s_\star$, then $\varphi_e(n)[t(e,s)]\downarrow$ for every $n \leq s$. In particular $\varphi_e(s) \leq d(s)$ for every $s \geq s_\star$. So d dominates φ_e.

Next assume there exists $d \leq_T A'$ that dominates every computable function. Define $h_s(e) = 1$ if $(\forall x \leq s)(\varphi_e(x)[d(s)]\downarrow)$ and 0 otherwise. It is easy to check that $\mathsf{Tot}(e) = \lim_s h_s(e)$. Hence Tot is limit computable in A and therefore $\mathsf{Tot} \leq_T A'$. So by Lemma 15.22, A is high. $\square$

Fact 15.24 (Ramsey Theorem). *Let $f : [\omega]^n \to K$ where $1 \leq K < \omega$. Then there exists an infinite $H \subseteq \omega$ such that $f \restriction [H]^n$ is constant. We call such H an f-homogeneous set.*

Lemma 15.25 (Jockusch). *There is a computable $f : [\omega]^2 \to \{0,1\}$ such that there is no infinite computable f-homogeneous set.*

Proof. Using Lemma 14.17, fix a 1-generic $X \subseteq \omega$ such that $X \leq_T 0'$. Note that $\omega \setminus X$ is also 1-generic. Hence both X and $\omega \setminus X$ are immune: For every $e < \omega$, if W_e is infinite, then $W_e \cap X \neq \emptyset$ and $W_e \cap (\omega \setminus X) \neq \emptyset$.

SInce $X \leq_T 0'$, it is limit computable. Let $f : \omega^2 \to \{0,1\}$ be a computable function such that $X(n) = \lim_s f(n,s)$. Now check that whenever H is infinite and c.e., $f \restriction [H]^2$ is not constant. $\square$

Fact 15.26 (Jockusch). *Let $f : [\omega]^2 \to 2$ be computable. Then there exists a Π^0_2 infinite f-homogeneous set.*

Theorem 15.27 (Jockusch). *There is a computable $f : [\omega]^2 \to \{0,1\}$ such that there is no infinite f-homogeneous set that is $\leq_T 0'$. Hence there is no Σ^0_2 infinite f-homgeneous set.*

Proof. The proof of Shoenfield limit lemma 15.11 gives a computable $g : \omega^3 \to \{0,1\}$ such that for every $A \leq_T 0'$, there exists $e < \omega$ such that $A(n) = \lim_s g(e,n,s)$.

Define $A_e = \{n : \lim_s g(e,n,s) = 1\}$ if $\lim_s g(e,n,s)$ exists for every n. Otherwise A_e is undefined. Define B_e to be the set of first $2e+2$ members of A_e (if A_e is defined and $|A_e| \geq 2e+2$, otherwise B_e is undefined). Define B^s_e to be the first $2e+2$ members of $\{n < s : g(e,n,s) = 1\}$ if $|\{n < s : g(e,n,s) = 1\}| \geq 2e+2$. Otherwise B^s_e is undefined. Note that $\langle B^s_e : e, s < \omega \rangle$ is computable.

Construct a computable partition $[\omega]^2 = X \sqcup Y$ as follows. $X = \bigcup_s X_s$ and $Y = \bigcup_s Y_s$ will be constructed in stages. At stage s, we will add each pair $\{n < s\}$ to $X_s \sqcup Y_s$. So $X_s \sqcup Y_s = \{\{n,s\} : n < s\}$.

Stage s: This stage is divided into $s+1$ substages $0 \leq e \leq s$. Note that at each substage $e < s$, we will add at most two pairs to $X \sqcup Y$.

Substage $e < s$: If B^s_e is undefined do nothing. Otherwise, let D^s_e be the set of those n such that $\{n, s\}$ is already in $X \sqcup Y$. Since at most two pairs entered $X \sqcup Y$ at each previous substage $e' < e$, we must have $|D^s_e| \leq 2e$. Since $|B^s_e| = 2e+2$, we can choose $n_1 < n_2$ in $B^s_e \setminus D^s_e$ and add $\{n_1, s\}$ to X and $\{n_2, s\}$ to Y.

Substage $e = s$: If some $\{n < s\}$ is not in $X \sqcup Y$, add it arbitrarily.

Now check that $f : [\omega]^2 \to \{0,1\}$ defined by $f(x) = 1$ iff $x \in X$ is as required.

$\square$