

**Exercise 11.5** Let  $x_1, \dots, x_n \in \mathbb{N}$ . For each of the following problems, either (a) design and analyze a polynomial time algorithm (the faster the better), or (b) prove that a polynomial time algorithm would imply a polynomial time algorithm for SAT.<sup>2</sup>

<sup>2</sup>You can use the solution of one subproblem to solve another, as long as there's no circular dependencies overall.

**Exercise 11.5.1.** The *partition problem* asks if one can partition  $x_1, \dots, x_n$  into two parts such that the sums of each part are equal.

*Solution.* We claim that a polynomial time solution for the partition problem would imply a polynomial time solution for SAT. To see this, we present a polynomial time reduction from subset sum, a problem known to be hard, to the partition problem.

**Notation.** Given some set  $S = \{s_1, \dots, s_n\}$ , we denote the sum  $\sum_{s \in S} s$  with  $\Sigma S$ .

Consider an arbitrary instance of subset sum. That is, suppose we have a set of positive integers

$$A := \{\alpha_1, \dots, \alpha_n\} \subseteq \mathbb{N}$$

and a positive integer target value  $T \in \mathbb{N}$ .

Now, let  $x := 2T - \Sigma A$  and define a new set  $\bar{A} := A \cup \{x\}$ . We claim that  $A$  has a subset sum to  $T$  if and only if an exact partition of  $\bar{A}$  exists.

Note that if  $T < \Sigma A/2$ , the value of  $x$  becomes less than 0. However, if there exists some set  $B = \{\beta_1, \dots, \beta_k\} \subseteq A$  such that  $\Sigma B = T$ , then  $A \setminus B$  sums to  $\Sigma A - T \leq \Sigma A/2$ . Thus, WLOG we can simply rephrase the problem to use  $\Sigma A - T$  as the target value instead.

Then, notice that

$$\begin{aligned} \Sigma \bar{A} &= \Sigma A + 2T - \Sigma A \\ &= 2T. \end{aligned}$$

*Correctness.* (SS  $\implies$  PP) Suppose there exists some  $B \subseteq A$  such that  $\Sigma B = T$ . Consider the partition of  $\bar{A}$  defined as  $\bar{B} = B \cup \{x\}$ . Then,

$$\Sigma \bar{B} = T + 2T - \Sigma A.$$

The remaining partition is then  $C := \bar{A} \setminus \bar{B}$ , and

$$\begin{aligned} \Sigma C &= 2T - T + 2T - \Sigma A \\ &= T + 2T - \Sigma A, \end{aligned}$$

and we can see that these two sums are equal. Hence, the partition problem is solved.

(SS  $\Leftarrow$  PP) Suppose the set  $\bar{A}$  has a valid partition such that each of the two subsets sum to  $T$ . Recall that  $\bar{A}$  is defined as the union of  $A$  and the singleton set  $\{x\}$ . By the pigeonhole principle, we know that one of these subsets of  $\bar{A}$  is a subset of  $A$ , whence the subset sum problem is solved. ■

Since each step in the reduction process takes only  $O(1)$  or  $O(n)$  time, the entire reduction can be done in polynomial time relative to the size of  $A$ . Thus, a polynomial time solution for the partition problem implies a polynomial time solution for SAT. □

**Exercise 11.5.2.** The *3-partition problem* asks if one can partition  $x_1, \dots, x_n$  into 3 parts such that the sums of each part are all equal.

*Solution.* We claim that a polynomial time solution for the 3-partition problem would imply a polynomial time solution for SAT. To see this, we present a polynomial time reduction from the partition problem, which we showed to be hard in 11.5.1.

**Notation.** Given sets  $A$  and  $B$ , we denote the *disjoint union* of  $A$  and  $B$  by  $A \sqcup B$ .

Consider an arbitrary instance of the partition problem. That is, consider a set of positive integers

$$A := \{\alpha_1, \dots, \alpha_m\} \subseteq \mathbb{N}.$$

The partition problem seeks two disjoint subsets  $B, C \subseteq A$  such that  $B \sqcup C = A$  and  $\Sigma B = \Sigma C$ .

Note that if  $\Sigma A$  is odd or the cardinality of  $A$  is less than 2, then the problem becomes impossible. Thus, WLOG we may assume that  $\Sigma A = 2n$  for some  $n \in \mathbb{N}$  and that  $A$  contains at least 2 elements.

Now, let  $x := n$  and define a new set  $\bar{A} := A \cup \{x\} \implies \Sigma \bar{A} = 3n$ .

*Correctness.* (PP  $\implies$  3P) Assume that  $\exists B \subseteq A$  such that  $\Sigma B = n$ . Let  $C := A \setminus B$  and notice that  $\Sigma C = 2n - n = n$ . By construction,

$$\bar{A} = C \sqcup B \sqcup \{x\} \text{ and } \Sigma C = \Sigma B = \Sigma \{x\}.$$

Hence, the 3-partition problem is solved.

(PP  $\Leftarrow$  3P) Assume that there exists a valid 3-partition of  $\bar{A}$ . That is, assume that there exist  $A_1, A_2, A_3 \subseteq \bar{A}$  such that

$$\Sigma A_1 = \Sigma A_2 = \Sigma A_3 = n \text{ and } A_1 \sqcup A_2 \sqcup A_3 = \bar{A}.$$

We already know  $\{x\} \subseteq \bar{A}$  and  $x = n$ , so WLOG we can set  $A_1 := \{x\}$ . Then, we have that  $A_2 \sqcup A_3 = \bar{A} \setminus A_1 = A$ , and we know  $\Sigma A_2 = \Sigma A_3 = n$ , whence the partition problem is solved. ■

This reduction can obviously be done in polynomial time relative to the size of  $A$ . Thus a polynomial time solution for the 3-partition problem would imply a polynomial time solution for the partition problem, which we have already shown would imply a polynomial time solution for SAT. □

**Exercise 11.5.3.** The *any- $k$ -partition problem* asks if one can partition  $x_1, \dots, x_n$  into  $k$  parts, for any integer  $k \geq 2$ , such that the sums of each part are all equal.

*Solution.* We claim that a polynomial time solution for the  $k$ -partition problem would imply a polynomial time solution for SAT. To see this, we present an inductive proof of a polynomial time reduction from the 3-partition problem to the  $k$ -partition problem.

As stated, our base case will be the 3-partition problem, which we showed to be hard in 11.5.2. Assume that we have shown that the  $\ell$ -partition problem is hard for all  $3 \leq \ell < k$ .

Consider an arbitrary instance of the  $(k-1)$ -partition problem. That is, consider a set of positive integers

$$A := \{\alpha_1, \dots, \alpha_m\} \subseteq \mathbb{N}.$$

The  $(k-1)$ -partition problem seeks  $k-1$  pairwise disjoint subsets  $A_1, A_2, \dots, A_{k-1} \subseteq A$  such that

$$\bigsqcup_{1 \leq i \leq k-1} A_i = A \quad \text{and} \quad \Sigma A_1 = \Sigma A_2 = \dots = \Sigma A_{k-1}.$$

Note that if  $\Sigma A$  is not divisible by  $k-1$  or if the cardinality of  $A$  is less than  $k-1$ , then the problem is rendered impossible. Thus, WLOG we may assume that  $\Sigma A = (k-1)n$  for some  $n \in \mathbb{N}$  and that  $A$  contains at least  $k-1$  elements.

By our inductive hypothesis, we have that the existence of a polynomial time solution for the  $(k-1)$ -partition problem implies the existence of a polynomial time solution for SAT.

Now, let  $x := n$  and define a new set  $\bar{A} := A \cup \{x\} \implies \Sigma \bar{A} = kn$ .

*Correctness.*  $((k-1)P \implies kP)$  Assume that there exists a valid  $(k-1)$ -partition for  $\bar{A}$ . That is, assume that there exist  $k-1$  pairwise disjoint subsets  $A_1, \dots, A_{k-1} \subseteq \bar{A}$  such that

$$\bigsqcup_{1 \leq i \leq k-1} A_i = \bar{A} \quad \text{and} \quad \Sigma A_1 = \Sigma A_2 = \dots = \Sigma A_{k-1}.$$

By construction, we have that

$$\bar{A} = \left[ \bigsqcup_{1 \leq i \leq k-1} A_i \right] \sqcup \{x\} \quad \text{and} \quad \Sigma A_1 = \Sigma A_2 = \dots = \Sigma A_{k-1} = \Sigma \{x\} = n.$$

Hence, the  $k$ -partition problem is solved.

$((k-1)P \Leftarrow kP)$  Assume that there exists a valid  $k$ -partition of  $\bar{A}$ . That is, assume there exist  $k$  pairwise disjoint subsets  $A_1, A_2, \dots, A_k \subseteq \bar{A}$  such that

$$\bar{A} = \bigsqcup_{1 \leq i \leq k} A_i \quad \text{and} \quad \Sigma A_1 = \Sigma A_2 = \dots = \Sigma A_k = n.$$

We already know  $\{x\} \subseteq \bar{A}$  and  $x = n$ , so WLOG we can set  $A_1 := \{x\}$ . Then, we have that

$$\bigsqcup_{2 \leq i \leq k} A_i = \bar{A} \setminus A_1 = A \quad \text{and} \quad \Sigma A_2 = \Sigma A_3 = \dots = \Sigma A_k = n,$$

whence the  $(k-1)$ -partition problem is solved. ■

This reduction can obviously be done in polynomial time relative to the size of  $A$ . Thus, a polynomial time solution for the  $k$ -partition problem implies a polynomial time solution for the  $(k-1)$ -partition problem, and by induction does so for the 3-partition problem (and equivalently for SAT). □

**Exercise 11.5.4.** The *almost-partition problem* asks if one can partition  $x_1, \dots, x_n$  into two parts such that the two sums of each part differ by at most 1.

*Solution.* We claim that a polynomial time solution for the almost-partition problem would imply a polynomial time solution for SAT. To see this, we present a polynomial time reduction from the partition problem, which we proved in 11.5.1 to be hard.

Suppose we want to solve the partition problem on a set of positive integers  $A := \{a_1, \dots, a_n\} \subseteq \mathbb{N}$ , given a solution to the almost-partition problem as a blackbox. We transform  $A$  into the set

$$A' := \{2a_i : a_i \in A\} = \{2a_1, \dots, 2a_n\}$$

and then apply the almost-partition solution to  $A'$ . Since all elements in  $A'$  are even, it is impossible for partitions to differ by exactly 1. Hence, we claim  $A$  has a partition if and only if  $A'$  has an almost-partition.

*Correctness.* To prove correctness, let us first assume that  $A'$  has an almost-partition; that is, there exists some  $B' \subseteq A'$  for which

$$\sum B' = \sum (A' \setminus B') \quad \text{or} \quad \sum B' = \sum (A' \setminus B') \pm 1$$

$B'$  and  $A' \setminus B'$  are both subsets of  $A'$ , so we have  $2 \mid \sum B'$  and  $2 \mid \sum (A' \setminus B')$ .

Since  $\sum B' = \sum (A' \setminus B') \pm 1$  cannot be true, we must have  $\sum B' = \sum (A' \setminus B')$ , which can be rewritten  $2 \sum B' = \sum A'$ .

Let  $B := \{a_i : 2a_i \in B'\} \subseteq A$ . Then  $2 \sum B = \sum B' = \frac{1}{2} \sum A' = \sum A$ ; hence,  $A$  has an exact partition.

Conversely, we now assume that  $A$  has an exact partition given by  $\sum B = \sum (A \setminus B)$  for some  $B \subseteq A$ . If we define  $B' := \{2a_i : a_i \in B\} \subseteq A'$ , then  $A'$  also has an exact partition given by  $\sum B' = \sum (A' \setminus B')$  which is, by definition, an almost-partition of  $A'$ . ■

This reduction can clearly be performed in polynomial time relative to the input size of  $A$  and expression size of the integers in  $A$ . Since we proved above that almost-partition can be used to solve the exact partition problem, which is known to be hard, we can conclude that a polynomial time solution for the almost-partition problem would also imply a polynomial-time solution for SAT. □

**Exercise 11.5.5.** <sup>3</sup>Let  $n$  be even. The *perfect partition problem* asks if one can partition  $x_1, \dots, x_n$  into two parts such that

- (a) Each part has the same sum.
- (b) Each part contains the same number of  $x_i$ 's.

<sup>3</sup>IMO, this one is the trickiest.

*Solution.* We claim that a polynomial time solution for the partition problem would imply a polynomial time solution for SAT. To see this, we present a polynomial time reduction from the partition problem, a problem known we showed to be hard in 11.5.1, to the perfect partition problem.

Consider an arbitrary instance of the partition problem. That is, consider a set of positive integers

$$A := \{\alpha_1, \dots, \alpha_m\} \subseteq \mathbb{N}.$$

The partition problem seeks two disjoint subsets  $B, C \subseteq A$  such that  $B \sqcup C = A$  and  $\Sigma B = \Sigma C = \Sigma A/2$ .

Note that if  $\Sigma A$  is odd or the cardinality of  $A$  is less than 2, then the problem becomes impossible. Thus, WLOG we may assume that  $\Sigma A = 2T$  for some  $T \in \mathbb{N}$  and that  $A$  contains at least 2 elements.

In any valid partition of  $A$ , the two parts may have different cardinalities. Suppose that in a given partition we have  $|B| = p$  and  $|C| = q$ . WLOG we can assume  $p \geq q$ , and let the difference between the cardinalities be  $x = p - q$ .

Note that if  $x = 0$ , then the partition is already 'perfect'. However if  $x > 0$ , although  $A$  is partitionable, it is not perfectly partitionable.

We can't yet know the value of  $x$ , but it is easy to see that it is bounded above by  $m$ . Then for each possible  $x \in \{1, \dots, m\}$ , define  $A_x = A \cup S_x \cup Y_x$  where  $S_x := \{s_1, \dots, s_{x+1}\}$  such that  $s = 1$  for each  $s \in S_x$  and  $Y_x := \{x+1\}$ .

*Correctness.* (PP  $\implies$  PPP) Assume that  $\exists B, C \subseteq A$  such that  $B \sqcup C = A$  and  $\Sigma B = \Sigma C = T$ . Let  $x = |B| - |C|$

*Case 1* ( $x = 0$ ). As above, this case is trivial. The partition is already perfect and hence the perfect partition problem is solved. ■

*Case 2* ( $x \geq 1$ ). Construct  $A_x$  as above. Now, consider the partition of  $A_x$  in which  $B_x := B \cup Y$  and  $C_x := C \cup S_x$ . Then  $\Sigma B_x = T + (x+1)$  and  $\Sigma C_x = T + (x+1)$ . Also,  $|B_x| = |B| + 1$  and  $|C_x| = |C| + (x+1)$ . However notice that by definition of  $x$ ,  $|C_x| = |C| + (|B| - |C| + 1) = |B| + 1 = |B_x|$  and thus the sets have equal cardinality. Hence the perfect partition problem is solved. ■

(PP  $\Leftarrow$  PPP) Assume that there exist disjoint  $B, C \subseteq A$  such that

$$B \sqcup C = A, \quad \Sigma B = \Sigma C = n, \quad \text{and} \quad |B| = |C|.$$

Obviously, the partition problem is solved. ■

Each step of the reduction process can be done in polynomial time relative to the size of  $A$ , so it follows that the a polynomial time solution to the perfect partition problem implies a polynomial time solution to the partition problem, whence a polynomial time solution to SAT. □