Note: There may be a version 2 coming.

Scope:

- DPV Chapter 4, Sections 4.1–4.5 and 4.7
- DPV Chapter 5, all sections
- DPV Chapter 6, all sections
- DPV Chapter 7, Sections 7.1–3 and 7.7
- DPV Chapter 8, all sections
- DPV Chapter 9, some terminology (see below)

Terminology

Be able to give the definition and/or an example of the following. (Page numbers are for the paper version of DPV.)

- approximation ratio (DPV, section 9.2)
- branch and bound (DPV, page 275)
- the boolean circuit value problem (DPV, page 221)
- a checking algorithm for a search problem (DVP, Chapter 8, page 234/258)
- the clique problem (DPV, page 242)
- conjuctive normal form (DPV, Chapter 8, page 234/249)
- decision problems (DPV, Chapter 8, page 258 in the *Why P and NP?* box)
- Euler path (DPV, page 100)
- optimization problems (page 188)
- Rudrata Path (DPV, page 265)
- Subset sum (DPV, page 242)
- the 3D Matching problem (DPV, page 240)
- 3SAT (DPV, page 235)
- the Traveling Salesman Problem (DPV, page 235)
- vertex cover (DPV, page 241)

Definitions/Examples will be about 15% of the points of the test.

Sample Problems

- Minimal-spanning-tree/shortest-paths problems:

  `http://www.cim.mcgill.ca/~langer/251/E8-MST.pdf`

  `https://www.lix.polytechnique.fr/~liberti/teaching/isic/isc612-07/ex-gph-sol.pdf`

- Dynamic programming practice problems:

  `https://blog.usejournal.com/top-50-dynamic-programming-practice-problems-4208fed71aa3`

- Greedy algorithm problems:

  `https://www.geeksforgeeks.org/greedy-algorithms/`

- Some NP-completeness problems (most too hard for the exam)

  `https://www.hse.ru/mirror/pubs/share/217960059`

Some LP and NP problems of the sort that might be on the exam

Problem 1 Express the following as a linear programming problem, specifically: *(i)* what are the variables, *(ii)* what is the objective function, *(iii)* what are the constraints? No need to solve the LP problem.

A farmer has 80 hectares of his farm available for planting maize and cabbages. He must grow at least 10 hectares of maize and 20 hectares of cabbages to meet demands. He prefers to plant more cabbages than maize but his work force and equipment will only allow him to cultivate a maximum of three times the quantity of cabbages to that of maize. His profit on maize is $800 per hectare and on cabbages $500 per hectare. How much of each crop should the farmer plant to maximize his profit?

Problem 2 BACKGROUND: See the definitions **Hitting Set** and **2-Hitting Set** on the reference page.

YOUR PROBLEM: Show **2-Hitting Set** $\trianglelefteq$ **Hitting Set**.

(Note: You may hand-wave on your f and h being poly-time—so long as they really are poly-time.)

Problem 3 Suppose X and Y are two search problems with $X \trianglelefteq Y$, i.e., X poly-time reduces to Y. For each of the following statements, say whether it is true or false and explain why.

(a) If Y is NP-complete, then X is also.
(b) If X is NP-complete, then Y is also.
(c) If Y is NP-complete and X is in NP, then X is NP-complete.
(d) If X is NP-complete and Y is in NP, then Y is NP-complete.
(e) If X is in P, then Y is also.
(f) If Y is in P, then X is also.

An answer to problem 1. See page 2 of http://www.durban.gov.za/Documents/City_Government/Maths_Science_Technology_Programme/mathematics-newsletter.pdf.

An answer to problem 2. Define:

$$f(\{ A_1, \ldots, A_k \}, b) = (\{ A_1, \ldots, A_k \}, b) \qquad h(\{ A_1, \ldots, A_k \}, b) = (\{ A_1, \ldots, A_k \}, b)$$

I.e., f and h do not change anything. Fix a particular instance of **2-Hitting Set** $(A'_1, \ldots, A'_n), b')$. Then for **(i)** through **(iv)** in the definition of $\trianglelefteq$:

- **(i)** Since **2-Hitting Set** is a restriction of **Hitting Set**, $(A'_1, \ldots, A'_n), b')$ is an instance of **Hitting Set**.
- **(ii)** If H is a set of size $\leq b_0$, then it works as a potential solution of $(A'_1, \ldots, A'_n), b')$ both as an instance of **Hitting Set** and **2-Hitting Set**.
- **(iii)** If there is an H of size $\leq b'$ such that, for all i, $H \cap A_i \neq \emptyset$, then it works as a solution of $(A'_1, \ldots, A'_n), b')$ both as an instance of **Hitting Set** and **2-Hitting Set**.
- **(iv)** This is just a repeat of the previous part.

Therefore, since f and h are polytime and satisfy (i), (ii), (iii), and (iv) of Definition 2, we have that **2-Hitting Set** $(A'_1, \ldots, A'_n), b')$.

Answer for problem 3.
(a) False
(b) False
(c) False
(d) True
(e) False
(f) True
We'll go over the *why*-part in class on Monday.

Some Reference Math Facts

a. $a^m \cdot a^n = a^{m+n}$.

b. $a^{m \cdot n} = (a^m)^n = (a^n)^m$.

c. $a^n \cdot b^n = (a \cdot b)^n$.

d. $\log_a a^n = n$.

e. $a^{\log_a n} = n$.

f. $c \cdot \log_a b = \log_a b^c$.

g. $\log_a (b \cdot c) = (\log_a b) + (\log_a c)$.

h. $\log_a x = (\log_a b) \cdot (\log_b x)$.

i. $a^{\log_b c} = c^{\log_b a}$.

j. $\sum_{k=1}^{n} k = \frac{n \cdot (n+1)}{2}$.

k. $\sum_{k=0}^{n} a^k = \frac{a^{n+1}-1}{a-1}$.

l. $\sum_{k=1}^{\infty} 2^{-k} = 1$.

m. $\sum_{k=1}^{n} k^2 = \frac{n \cdot (n+1) \cdot (2n+1)}{6}$.

n. $\sum_{k=1}^{\infty} k \cdot 2^{-k} = 2$.

o. For $a > 1$ and $b, c > 0$:

$$\lim_{n\to\infty} \frac{(\log_a n)^b}{n^c} = 0.$$

$$\lim_{n\to\infty} \frac{n^c}{(\log_a n)^b} = \infty.$$

p. For $a > 1$ and $b, c > 0$:

$$\lim_{n\to\infty} \frac{n^b}{a^{c \cdot n}} = 0.$$

$$\lim_{n\to\infty} \frac{a^{c \cdot n}}{n^b} = \infty.$$

q. If $\lim_{n\to\infty} \frac{f(n)}{g(n)} = c$, then:

	$c = 0$	$0 < c < \infty$	$c = \infty$
$f(n) \in O(g(n))$	True	True	False
$f(n) \in \Theta(g(n))$	False	True	False
$f(n) \in \Omega(g(n))$	False	True	True

Dijkstra's Algorithm. GIVEN: $G = (\{1, \dots, n\}, E)$ and a function len that, for each $u, v \in \{1, \dots, n\}$:

(i) if $(u, v) \in E$, then $len(u, v) > 0$, and
(ii) if $(u, v) \notin E$, then $len(u, v) = \infty$.

GOAL: For each $i \in \{2, \dots, n\}$, compute $dist[i]$ = the length of a *shortest* path from 1 to i.

```
function Dijkstra(G, ℓ)
  array dist[1..n]
  ToDo ← { 2, . . . , n }
  for i ← 2 to n do dist[i] ← len(1, i)
  for i ← 1 to n − 1 do
    choose v ∈ ToDo with minimal dist[v]
    ToDo ← ToDo − { v }
    for each w adjacent to v do
      dist[w] ← min(dist[w], dist[v] + len(v, w))
  return dist
```

Definition: CNF-Terminology.

(i) The negation of a boolean variable x is written: $\overline{x}$.
(ii) A *literal* is either a boolean variable or the negation of a boolean variable.
(iii) A *clause* is a disjunction of literals. E.g.: $x \vee \overline{y} \vee \overline{z}$.
(iv) A *conjunctive normal form (CNF)* clause is a conjunction of clauses. E.g.: $(x \vee \overline{y} \vee \overline{z}) \wedge (\overline{w} \vee z) \wedge (u \vee \overline{x} \vee x)$.

Definition: Reducibility.
For search problems P and Q, $P \trianglelefteq Q$ means that there are polynomial-time computable functions f and h such that (i) through (iv) hold.

(i) If I is an instance of P, then $f(I)$ is an instance of Q.
(ii) If S is a potential solution of $f(I)$, then $h(S)$ is a potential solution of I.
(iii) If I is an instance of P that has a solution, then $f(I)$ is an instance of Q that has a solution.
(iv) If $f(I)$ is an instance of Q that has a solution S, then $h(S)$ is a solution of instance I of P.

Definition. $\mathbb{N} = \{0, 1, 2, \dots\}$ and $\mathbb{N}^+ = \{1, 2, 3, \dots\}$.

Definition: The Hitting-Set Problem.
GIVEN $b \in \mathbb{N}^+$ and a family of sets $\mathcal{S} = \{A_1, A_2, \dots, A_n\}$,
FIND: a set H of size $\le b$ such that $(\forall A \in \mathcal{S})[H \cap A \neq \emptyset]$.

Definition: The 2-Hitting-Set Problem.
GIVEN: $b \in \mathbb{N}^+$ and a family of sets $\mathcal{S} = \{A_1, A_2, \dots, A_n\}$ where each A_i is of size 2,
FIND: a set H of size $\le b$ such that $(\forall A \in \mathcal{S})[H \cap A \neq \emptyset]$.