

2019

SMA KOMPUTER



085223273373

PEMBAHASAN PAKET 9

1. Soal ini adalah soal bertipe *dynamic programming*.
Misalkan $F(x)$ adalah banyaknya cara Pak Dengklek memindahkan x buah kardus miliknya dari kantor yang lama ke kantor yang baru.
Jelas bahwa $F(1) = 1$. Kemudian untuk $F(2)$, Pak Dengklek bisa membawa dua kardus tersebut secara langsung atau membawa nya satu per satu sehingga terdapat 2 cara ($F(2) = 2$)
Jika banyaknya kardus yang dimiliki Pak Dengklek ≥ 3 . Pak Dengklek tentu memiliki 2 buah opsi.
 - Opsi pertama, Pak Dengklek membawa 1 kardus terlebih dahulu. Hal ini mengakibatkan banyaknya cara adalah $F(x-1)$.
 - Opsi kedua, Pak Dengklek membawa 2 buah kardus terlebih dahulu. Hal ini mengakibatkan banyaknya cara adalah $F(x-2)$.Sehingga, dari sini kita mendapatkan bahwa untuk $x \geq 3$, maka $F(x) = F(x-1) + F(x-2)$.
 $F(1) = 1$, $F(2) = 2$, lalu $F(x) = F(x-1) + F(x-2)$.
Dengan basis dan rekurens ini, kita mampu mendapatkan bahwa nilai dari $F(13)$ adalah 377
Jawaban : **E**
2. Agar jumlahnya maksimal, maka bilangan yang diambil adalah 3 bilangan yang paling besar. Jumlahnya adalah $19 + 12 + 11 = 42$
Jawaban : **D**
3. Agar hasil perkaliannya maksimal, maka bilangan yang harus Andi pilih juga sebesar mungkin. Bilangan itu adalah 23
Jawaban : **E**
4. Strategi yang paling optimal adalah memenuhi 6 kandang dengan masing-masing berisi 10 bebek dan sisa 40 bebek disebar kedalam 15 kandang. Dari 15 kandang untuk 40 bebek sisa tersebut dijamin tidak akan ada yang padat penduduk. Sehingga minimal kandang yang padat penduduk adalah 6.
Jawaban : **D**
5. Keuntungan maksimal yang dapat diperoleh adalah $(51.000 - 20.000) \times 4 = 124.000$
Jawaban : **D**
6. Total kebahagiaan yang bisa diperoleh adalah $9 \times 8 + 5 \times 7 + 2 \times 0 + (-3)(-1) + (-6)(-4) = 134$
Jawaban : **A**
7. Permasalahan ini adalah permasalahan coin change dengan *dynamic programming*. Banyaknya minimal koin yang diperlukan adalah 5. (3 koin nominal 4 dan 2 koin nominal 3)

Jawaban : **B**

8. Banyaknya koin minimal adalah 5. 4 koin dengan nominal 4 dan 1 koin dengan nominal 2.

Jawaban : **D**

9. Permasalahan ini dapat diselesaikan dengan greedy, yaitu Ambil 2^x terbesar yang mungkin. Dengan cara ini kita mendapatkan $2018 = 2^{10} + 2^9 + 2^8 + 2^7 + 2^6 + 2^5 + 2^1$

Sehingga n minimum adalah = 7

Jawaban : **D**

10. Cara yang mirip dengan nomor 9. Karena $2018 = 2 * 3^6 + 2 * 3^5 + 2 * 3^3 + 2 * 3^2 + 2 * 3^0$, maka nilai n minimum adalah $2 + 2 + 2 + 2 + 2 = 10$

Jawaban : **C**

11. Permasalahan ini adalah permasalahan *dynamic programming*.

Misalkan $F(N)$ adalah banyaknya cara konfigurasi ubin dengan ukuran $3 \times N$.

Perhatikan bahwa seseorang dapat menutup kolom pertama ubin dengan 2 cara:

- Menutup kolom pertama ubin dengan ubin 1×3 secara vertikal. Hal ini mengakibatkan banyaknya cara untuk menutupi sisa ubin tersebut adalah $F(N-1)$
- Menutup kolom pertama ubin dengan 3 ubin 1×3 secara horizontal. Hal ini mengakibatkan banyaknya cara untuk menutupi sisa ubin tersebut adalah $F(N-3)$

Sehingga kita mendapatkan relasi $F(N) = F(N-1) + F(N-3)$. Basis untuk kasus ini adalah $F(1) = 1$, $F(2) = 1$, $F(3) = 2$. Dari sini kita mendapatkan bahwa nilai dari $F(6) = 6$

Jawaban : **C**

12. Dengan menggunakan data yang ada pada nomer sebelumnya, maka kita mendapatkan nilai dari $F(10)$ adalah 28

Jawaban : **E**

13. Agar nilai sum menjadi terkecil, maka kita harus memasang bilangan tersebut dengan aturan $(\max_di_array_A) * (\min_di_array_B)$

Sehingga jika array $A = \{3, 1, 9, 2\} = \{1, 2, 3, 9\}$ dan $B = \{2, 1, 4, 5\} = \{1, 2, 4, 5\}$

Maka pemasangan yang optimal adalah $9 \times 1 + 3 \times 2 + 2 \times 4 + 1 \times 5 = 9 + 6 + 8 + 5 = 28$

Jawaban : **B**

14. Dengan menggunakan prinsip yang sama seperti pada soal nomor 13, maka kita akan mendapatkan nilai sum = $1 \times 20 + 2 \times 19 + 3 \times 18 + \dots + 20 \times 1$

Nilai ini sama dengan 1540

Jawaban : **C**

15. Longest increasing subsequence adalah sebuah permasalahan yang cukup terkenal dalam *dynamic programming*. Longest Increasing Subsequence dari array tersebut adalah 6, yaitu : $\{1, 2, 3, 5, 6, 7\}$

Jawaban : **D**

16. Banyaknya cara perpindahan dari Kotak A ke kotak lain dapat dituliskan ke dalam tabel berikut: (cara berikut adalah cara *dynamic programming*)

A	1	1	1
1	2	3	4
1	3	6	10
1	4	10	20
1	5	15	B 35

Sehingga banyak cara seseorang berpindah dari kotak A ke kotak B dengan hanya berjalan ke kanan atau ke bawah saja ada 35 cara.

Jawaban : **C**

17. Dengan cara yang mirip pada nomor 16, kita bisa menuliskan banyaknya cara ke dalam tabel berikut:

A	1	1	1	1
1	1	0	1	2
0	1	1	2	0
0	1	2	4	4
0	0	2	6	10
0	0	2	0	B 10

Sehingga banyaknya cara adalah 10

Jawaban : **B**

18. Kita dapat menggunakan Greedy. Agar menghasilkan total selisih minimal, selisihkan bilangan yang bersebelahan.

Jawaban: **D**

19. Kita dapat menggunakan Greedy. Agar menghasilkan total selisih maksimal, selisihkan bilangan yang berjauhan.

Jawaban : **B**

20. Perhatikan apabila kita bagi bilangan tersebut menjadi 2 buah kelompok, kiri dan kanan. Kelompok kiri adalah 3, 4, 5, dan 9. Sedangkan kelompok kanan adalah 12, 17, 18, dan 20. Apabila kita memasangkan 1 bilangan di kiri dan 1 bilangan di kanan secara acak dan mencari total selisihnya, kita akan tetap mendapatkan jawaban 46. Sehingga jawabannya adalah $4! \cdot (2^4)$ ($4!$ dari banyaknya cara permutasi 4 buah bilangan, 2^4 karena (A,B) dan (B,A) dianggap pasangan yang berbeda).

Jawaban : **E**

21. Penekanan tombol minimal yang dapat dilakukan adalah 4. (2xmerah + 2xputih)

Jawaban : **B**

22. Salah satu cara adalah dengan melangkah mundur, bagi 2 atau tambah 1

$100 \rightarrow 50 \rightarrow 25 \rightarrow 26 \rightarrow 13 \rightarrow 14 \rightarrow 7 \rightarrow 8 \rightarrow 4 \rightarrow 5$

Jawaban : **C**

23. Salah satu cara adalah dengan melangkah mundur, bagi 2 atau tambah 1
1000 → 500 → 250 → 125 → 126 → 63 → 64 → 32 → 16 → 8 → 4 → 5

Jawaban : **C**

24. Untuk soal nomor 24 dan 25. Sebenarnya hanya ada 2 kemungkinan jawaban, yaitu n atau $n-1$. Soal yang jawabannya adalah n ketika jumlah dari arraynya habis dibagi n , dan selain itu maka jawabannya adalah $n-1$. Sehingga untuk nomor 24, karena jumlah bilangan pada arraynya $2-7-2-6 = -13 \neq 0 \pmod{4}$, maka jawabannya adalah 3

Jawaban : **C**

25. Dengan cara yang sama seperti pada nomor 24, jumlah dari array Pak Dengklek
 $= 1 + 3 + 2 + 9 - 9 + 2 + 10 + 13 + 21 = 52$, karena $52 \neq 0 \pmod{9}$, maka jawabannya adalah 8

Jawaban : **C**

26. Kita dapat menyelesaikan soal ini dengan Dynamic Programming. Pertama, kita harus

mencari persegi panjang dengan ukuran $1 \times P$ yang tidak dapat dipecah lagi dengan persegi panjang yang lebih kecil. Kemudian, kita bisa buat rekurens dari Dynamic Programming yang ada. Apabila Anda sudah mencari persegi panjang dengan benar, maka seharusnya Anda akan menemukan rekurensnya $f(x) = 2 * (f(x-1) + f(x-2) + \dots + f(0))$.

Jawaban : **B**

27. Sama seperti penjelasan pada nomor 26

Jawaban : **B**

28. Kita dapat mencari poin maksimum yang dapat Budi peroleh hingga mencapai kotak tertentu dalam tabel berikut: (menggunakan $f(x,y) = \max(f(x,y-1), f(x-1,y-1), f(x+1,y-1)) + \text{poin}[x][y]$)

3	4	9	18
2	7	17	22
5	8	18	20

Sehingga nilai maksimum yang dapat Budi peroleh adalah 22

Jawaban : **E**

29. Sama seperti nomor 28, kita dapat membuat nilai maksimum yang dapat Budi peroleh hingga mencapai kotak tertentu dalam tabel berikut:

3	7	9	12	16
2	8	10	15	18
5	6	11	14	17
4	7	10	13	15
1	9	11	12	17

Sehingga nilai maksimum yang dapat Budi peroleh adalah 18

Jawaban : **E**

30. Didefinisikan $D(n) = \text{true}$, jika Pak Dengklek bisa memenangkan pertandingan. Sebaliknya $D(n) = \text{false}$, jika Pak Dengklek tidak bisa memenangkan pertandingan. Karena Pak Dengklek memulai permainan, maka

$D(1) = \text{true}$, ambil 1 buah batu

$D(2) = \text{true}$, ambil 2 buah batu

$D(3) = \text{true}$, ambil langsung 3 batu

$D(4) = \text{false}$, karena Pak Dengklek tidak dapat langsung mengambil 4 batu, dan :

- Jika Pak Dengklek mengambil 1 batu, Pak Ganesh akan mengambil 3 batu dan menang
- Jika Pak Dengklek mengambil 2 batu, Pak Ganesh akan mengambil 2 batu dan menang
- Jika Pak Dengklek mengambil 3 batu, Pak Ganesh akan mengambil 1 batu dan menang

Kita dapat mendapatkan nilai $D(n)$ dengan melihat ke nilai $D(n - \text{fib})$ dimana fib adalah bilangan fibonacci. Anggap bahwa Pak Dengklek mengambil batu sebanyak fib, sehingga $D(n - \text{fib})$ mengacu ke strategi menang-kalah dari Pak Ganesh. Jika ada salah satu saja dari $D(n-1)$, $D(n-2)$, $D(n-3)$, $D(n-5)$, ... yang bernilai false, maka $D(n)$ bernilai true. Misalnya $D(5) = \text{true}$, karena $D(5-1)=\text{false}$, (1 adalah bilangan fibonacci)

Akan didapatkan $D(20) = \text{false}$. Sehingga Pak Dengklek pasti kalah. Pemenangnya adalah Pak Ganesh.

Jawaban : **A**