

Perhatikan potongan program di bawah ini untuk nomor 16 – 18!

```
int bebek(string S, int x, int y){
    if (lo == hi)
        return 1;
    if (S[x] == S[y] && x + 1 == y)
        return 2;
    if (S[x] == S[y])
        return bebek(S, x + 1, y - 1) + 2;
    return max(bebek(S, x, y - 1), bebek(x, x + 1, y));
}
```

1. Jika dipanggil bebek ("QWERTYTRL PQ", 0, 10) berapa nilai kembaliannya? **{tuliskan jawaban berupa ANGKA saja}**

JAWABAN : 7

Pembahasan:

Perhatikan bahwa fungsi bebek sebenarnya menghitung *subsequence* terpanjang S dari indeks x sampai y yang merupakan suatu palindrom (*string* yang sama apabila dibaca dari depan ke belakang atau belakang ke depan).

Subsequence terpanjang dari QWERTYTRL PQ yang merupakan palindrom adalah QWERTYTRL PQ sepanjang 7.

2. Untuk suatu string S sepanjang 3 karakter ada berapa banyak kemungkinan sehingga hasil pemanggilan fungsi bebek (S, 0, 2) bernilai maksimum? **{tuliskan jawaban berupa ANGKA saja}**

JAWABAN : 676

Catatan: Diasumsikan *string* S hanya boleh berisi huruf besar Latin.

Pembahasan:

Palindrom yang panjangnya 3 pasti berbentuk xyx, dengan x dan y merupakan suatu huruf. Ada 26 cara masing-masing untuk memilih x dan y, sehingga ada $26^2=676$ cara.

3. Manakah di bawah ini yang mempunyai nilai kembalian paling kecil dari hasil pemanggilan fungsi?
- bebek ("NGASALAJA", 0, 8)
 - bebek ("WOILAHCIK", 0, 8)
 - bebek ("NGAWII", 0, 4)
 - bebek ("JMK4848", 0, 6)
 - bebek ("GATAULAGIDAHMAUDIBIKINGIMANA", 0, 27)

JAWABAN C

Perhatikan bahwa pada pilihan C, kembaliannya adalah *subsequence* palindrom terpanjang dari NGAWI saja (bukan NGAWII, karena y = 4), yaitu 1. Pada pilihan lain, kita pasti bisa mendapatkan palindrom sepanjang 2 atau lebih (ada setidaknya 2 huruf yang sama).

Perhatikan potongan program di bawah ini untuk nomor 19 – 20!

```
int asam(int x){
    int y = 1/(x - 5);
    y *= x;
    y += x;
}
```

```

    return y + 3;
}
int manis(int x){
    return asam(x + 1) - asam (x - 1) + x;
}

```

4. Jika dipanggil fungsi asam (5) berapakah nilai kembaliannya?
- 6
 - 7
 - 8
 - Program gagal dikompilasi
 - Program dijalankan namun memberikan output yang tidak sesuai

JAWABAN E

Jika menjawab C relatif , program masih bisa dirun dan dikompilasi karena

5. Jika fungsi di atas dipanggil dan dijalankan pada program di bawah ini

```

int main(){
int sum = 0;
for(int i = 1; i<=1000; i++){
    if(manis(i) % 2 == 0){
        sum++;
    }

    if(manis(i) % 3 == 0){
        sum++;
    }
}
cout<<sum<<endl;
return 0;
}

```

Berapakah nilai keluaran yang dihasilkan? **{tuliskan jawaban dalam bentuk ANGKA saja}**

JAWABAN : 834

Anda bisa melakukan pengamatan pola untuk setiap nilai i kelipatan 2 atau kelipatan 3

manis(1) = 3

manis(2) = 4

manis(3) = 1

manis(4) = 6

manis(5) = 17

manis(6) = 8

manis(7) = 3

manis(8) = 10

manis(9) = 11

manis(10)= 12

manis(11) = 13

...

Dst

Dapat dipastikan bahwa untuk $x > 7$ dipastikan $\text{manis}(x) = x + 2$

- Untuk bilangan habis dibagi 2 :

Pada rentang $1 \leq x \leq 7$ ada sebanyak 3

Untuk $x > 7$ $\text{manis}(x)$ akan bernilai genap saat x nya juga genap, untuk $7 < x \leq 1000$, ada sebanyak $1000 / 2 - 3 = 497$

- Untuk bilangan habis dibagi 3 :

$\text{manis}(x)$ akan bernilai kelipatan 3 saat x bernilai $\{1, 4, 7, 10, 13, \dots, 1000\}$

$$U_n = 1 + (n - 1)3$$

$$U_n = 1 + 3n - 3 = 3n - 2$$

$$3n - 2 = 1000$$

$$3n = 1002$$

$$n = 1002 / 3$$

$$n = 334$$

ada sebanyak 334.

Nilai akhir sum = $3 + 497 + 334 = 834$.

6. Jika pemanggilan fungsi $\text{manis}(x + 5) = 2027$ tentukan berapa nilai x yang memenuhi **{tuliskan jawaban dalam bentuk ANGKA saja}**

JAWABAN : 2020

$\text{manis}(x) = x + 2$ jika $x > 7$, $x + 5 + 2 = 2027$, $x = 2020$

Perhatikan potongan program di bawah ini untuk nomor 22 – 23!

```
void crack(int n, int pos){
    if(n == 0){
        cout<<" ";
    }else if(n % 2 == 0 && n != 0){
        pos++;
        crack(n / 2, pos);
    }else{
        cout<<pos<<" ";
        pos++;
        crack(n / 2, pos);
    }
}
```

7. Jika dipanggil fungsi `crack(13, 0)` bagaimana tampilan keluaran yang dihasilkan?

- a. 1 2 3
- b. 0 2 3
- c. 1 3 2
- d. 3 2 1
- e. 0 3 2

JAWABAN B

Pembahasan : fungsi tersebut mengembalikan pada indeks ke berapa saja bit 1 berada pada representasi biner bilangan n .

8. Berapa hasil penjumlahan angka yang ditampilkan output pada pemanggilan fungsi `crack(1,0) + crack(2,0) + crack(22,0) + crack(23,0) + ... + crack(230,0)` {tuliskan jawaban berupa ANGKA saja}

JAWABAN : 465

Pembahasan :

`crack(2n,0)` = akan mengembalikan posisi bit 1 setelah dilakukan right shift sebanyak n kali

`crack(1,0) -> 1`

`crack(2,0) -> 01`

`crack(4,0) -> 001`

`crack(8,0) -> 0001`

`crack(1,0) + crack(2,0) + crack(22,0) + crack(23,0) + ... + crack(230,0) = 1 + 2 + 3 + 4 + ... + 30 = 465`

Perhatikan potongan program berikut untuk nomor 24 – 25!

```
void fly(string s, int x)
{
    if (x == s.size() - 1) {
        cout << s ;
        return;
    }
    for (int i = idx; i < s.size(); i++) {
        swap(s[x], s[i]);
        fly(s, idx + 1);
        swap(s[x], s[i]);
    }
}
```

9. Tentukan keluaran dari hasil pemanggilan `fly("AKU",0)`! {tuliskan jawaban dalam bentuk string tanpa dipisahkan spasi}

Jawaban: AKUAUKKAUKUAUKAUAK

Pembahasan:

Misalkan n merupakan panjang dari s. Perhatikan bahwa saat dipanggil, flys, x akan melakukan rekursi sebanyak n-x kali dan x bertambah 1. Maka, akan ada rekursi sebanyak n! kali.

Karena n!=6, fungsi rekursif tersebut cukup cepat untuk disimulasikan dan didapat hasilnya.

10. Jika untuk sebuah string kata = "SIGMA" dijalankan ke prosedur `fly(kata,0)` maka nilai akhir kata adalah ...

- AGIMS
- AMIGS
- ASIGM

- d. SIGMA
- e. SGIMA

JAWABAN D

Semua operasi swap yang dilakukan pada s (sebelum rekursi) pasti akan dikembalikan (setelah rekursi), sehingga nilai akhirnya pasti sama dengan nilai awal.

Perhatikan potongan program berikut untuk nomor 26 – 27

```
int main(){
    int t;
    cin>>t;
    int x,y;
    vector<int> angka;
    while (t--){
        cin>>x>>y;
        if(x > y){
            angka.push_back(x);
            angka.push_back(y);
        }else if(x == y){
            angka.push_back(x);
        }else{
            angka.push_back(y);
            angka.push_back(x);
        }

    }
    for(int number : angka){
        cout<<number<<" ";
    }
    return 0;
}
```

11. Jika keluaran program berupa sekumpulan angka 1 0 0 3 2 4 2 masukan mengikuti format input dari program yang TIDAK mungkin di bawah ini?
- a. 4
 - 10
 - 00
 - 32
 - 24

b. 5
1 1
0 0
0 0
3 2
2 4

c. 4
1 0
0 3
2 2
2 4

d. 5
1 1
0 0
0 0
3 3
2 2
4 2

e. 5
1 1
0 0
0 0
3 2
4 2

JAWABAN C

Untuk suatu baris masukan, jika kedua bilangannya berbeda, pasti dikeluarkan bilangan yang lebih besar terlebih dahulu. Pada baris kedua pilihan C, 0 3 akan ditukar saat dikeluarkan, menjadi 3 0.

12. Jika keluaran program menghasilkan 5 angka berbeda, berapa nilai t terbesar dan t terkecil yang dapat menjadi nilai masukan? **{Tuliskan jawaban berupa nilai t terbesar dan t terkecil secara berurutan dipisahkan oleh spasi}**

JAWABAN : 5 3

t terbesar = 5 yaitu saat masing – masing x pada iterasi t bernilai sama

a a

b b

c c

d d

e e

t terkecil yaitu = 3 saat masing – masing nilai x dan y bernilai sama kecuali pada salah satu iterasi

a b

c c

d e

Perhatikan potongan program berikut untuk nomor 28 – 30

```
int lempar(int n){
    if(n == 0) return 0;
    if(n == 1 || n == 5 || n == 20 || n == 100) return 1;
    if(n >= 100) return lempar(n%100) + n / 100;
    if(n >= 20) return lempar(n%20) + n / 20;
    if(n >= 10) return lempar(n%10) + n / 10 ;
    if(n >= 5) return lempar(n%5) + n / 5;
    if(n >= 1) return lempar(n%1) + n / 1;
}
```

13. Jika dipanggil fungsi `lempar(5000)` tentukan nilai kembaliannya **{Tuliskan jawaban dalam bentuk ANGKA saja}**

JAWABAN : 50

Function `lempar(n)` merupakan solusi abstrak untuk permasalahan menentukan lembarang minimal untuk membentuk uang nominal sebesar n dengan pecahan 1,5,20, dan 100, untuk membentuk 5000 minimal membutuhkan 50 lembar pecahan 100.

14. Tentukan nilai kembalian dari pemanggilan fungsi `lempar(812984)` **{Tuliskan jawaban dalam bentuk ANGKA saja}**

JAWABAN : 8137

$812984 = 100 * 8129 + 20 * 4 + 1 * 4$; total lembarang = $8129 + 4 + 4 = 8137$

15. Manakah di bawah ini pernyataan yang benar?

- a. `lempar(1713) == lempar(998)`
- b. `lempar(1620) == lempar(1700)`
- c. `lempar(687) == lempar(1620)`
- d. `lempar(893) == lempar(1713)`
- e. `lempar(556) == lempar(320)`

JAWABAN B

Dengan melakukan pengujian satu persatu dengan metode di atas anda akan menemukan jawabannya adalah B. Gunakan teknik Binary Search the Answer, sebagai contoh tidak mungkin `lempar(1713) == lempar(998)` karena otomatis banyak lembarang 100 yang dibutuhkan pasti berbeda.