

- **PROSEDUR, FUNGSI, DAN PEMROSESAN TEKS**

1. Buat program untuk mengkonversi suhu dari derajat Celcius ke Fahrenheit dan Kelvin. Program harus meminta suhu dalam Celcius dari pengguna, kemudian menampilkan suhu yang setara dalam Fahrenheit dan Kelvin. Buat kedalam fungsi dan procedure, dimana:

Fungsi konversiKeFahrenheit untuk mengkonversi suhu ke fahrenheit.

Prosedur tampilkanKelvin untuk menghitung suhu dalam Kelvin.

Gunakan rumus:

- Fahrenheit = (Celsius * 9/5) + 32
- Kelvin = Celsius + 273.15

Jawaban:

```
1  #include<stdio.h>
2
3  float konversiFahrenheit(float Celcius){
4      return (Celcius * 9/5) + 32;
5  }
6
7  void tampilkanKelvin(float Celcius){
8      printf("Suhu dalam Kelvin: %.2f\n", Celcius + 273.15);
9  }
10
11 int main(){
12
13     float Celcius;
14     float hasil;
15
16     printf("Masukkan suhu: ");
17     scanf("%f", &Celcius);
18
19     hasil = konversiFahrenheit(Celcius);
20     printf("Suhu dalam Fahrenheit: %.2f\n", hasil);
21     tampilkanKelvin(Celcius);
22
23     return 0;
24 }
```

2. Pak Dengklek yang sangat menyukai matematika sedang mengajarkan faktorial kepada Ucil, bebek Pak Dengklek. Karena Ucil sangatlah cerdas, ia pun dengan mudahnya mengerti faktorial. Oleh karena itu, Pak Dengklek memberikan tantangan baru kepada Ucil untuk mempelajari faktorial ganjil-genap.

Perhitungan faktorial ganjil-genap sama seperti faktorial biasa, hanya saja semua bilangan genap yang dikalikan harus dibagi dengan 2 terlebih dahulu. Seperti kita ketahui bahwa notasi dari faktorial untuk bilangan bulat N adalah $N!$. Sedangkan, notasi dari faktorial ganjil-genap untuk N adalah $N!!$.

Perhitungan dari $N!$ adalah sebagai berikut:

$$N!$$

$$= N \times (N-1)!$$

$$= N \times (N-1) \times (N-2) \times \dots \times 1$$

Sedangkan, perhitungan dari $N!!$ adalah sebagai berikut:

$$N!!$$


$$= f(N) \times (N-1)!!$$

$$= f(N) \times f(N-1) \times f(N-2) \times \dots \times f(1)$$

dengan $f(x) = x$ apabila x ganjil, atau $x/2$ apabila x genap. Sebagai contoh, $5!!$
 $= 5 \times 4/2 \times 3 \times 2/2 \times 1 = 30$.

Ucil diberikan sebuah bilangan bulat N . Bantulah Ucil membuatkan program untuk menghitung faktorial ganjil genap dari N . Khusus untuk program ini, Ucil meminta agar Anda menggunakan rekursi.

Jawaban:



```

1  #include<stdio.h>
2
3  int faktorial (int N){
4      if (N == 0)
5          return 1;
6      else if (N % 2 == 0)
7          return N / 2 * faktorial(N-1);
8      else
9          return N * faktorial(N-1);
10 }
11
12 int main(){
13     int N;
14     printf("Masukkan bilangan: ");
15     scanf("%d", &N);
16
17     faktorial(N);
18     printf ("N!! = %d", faktorial(N));
19
20     return 0;
21 }

```

- **ARRAY**

1. Diberikan sebuah array yang berisi N bilangan bulat. Elemen array tersebut akan diinputkan oleh user. Hitung berapa banyak elemen yang bernilai genap dan berapa banyak yang bernilai ganjil.

Input:

- Integer N yang menunjukkan ukuran array.
- N bilangan bulat sebagai elemen array.

Output:

- Dua angka, jumlah elemen genap dan jumlah elemen ganjil dalam array.

Jawaban:

```
1  #include<stdio.h>
2
3  int main(){
4
5      int N;
6      int ganjil=0, genap=0;
7      int array[N];
8
9      printf("Masukkan jumlah elemen: ");
10     scanf("%d", &N);
11
12     for (int i = 0; i < N; i++){
13         printf("Masukkan elemen ke-%d: ", i+1);
14         scanf("%d", &array[i]);
15
16         if (array[i] % 2 == 0){
17             genap += 1;
18         }else{
19             ganjil += 1;
20         }
21     }
22     printf ("Jumlah bilangan genap: %d\n", genap);
23     printf ("Jumlah bilangan ganjil: %d\n", ganjil);
24
25
26     return 0;
27 }
```

2. Gema dan Astik sedang mengikuti lomba pemrograman di Semarang. Sebelum pulang, Gema membelikan Astik oleh-oleh berupa lumpia rebung sepanjang L cm.

Karena Astik mempunyai N teman, maka Astik ingin memotong lumpia menjadi $N + 1$ bagian (1 bagian untuk dirinya sendiri) dengan memotong N kali. Pada mulanya, lumpia tersebut terdiri atas 1 bagian sepanjang L cm. Pada pemotongan ke- i , Astik memotong bagian ke- B_i dari kiri menjadi 2 bagian sama panjang. Dijamin bagian tersebut mempunyai panjang bilangan genap dalam satuan cm.

Kini Gema penasaran, berapa panjang masing-masing bagian setelah dipotong oleh Astik. Bantulah Gema menghitungnya!

Format Masukan dan Keluaran

Baris pertama masukan terdiri dari dua buah bilangan N dan L yang menyatakan banyaknya teman Astik dan panjang lumpia rebung dalam cm. Baris kedua terdiri dari N buah bilangan yang menyatakan nilai B_1, B_2, \dots, B_N ($1 \leq B_i \leq L$).

Keluaran terdiri dari sebuah baris berisi $N + 1$ bilangan yang menyatakan panjang masing-masing bagian dari kiri sampai kanan dalam cm.

Contoh Masukan/Keluaran

Masukan	Keluaran
4 1000 1 2 2 1	250 250 125 125 250

Penjelasan:

- Pada mulanya, lumpia terdiri dari 1 bagian sepanjang 1000 cm
- Setelah dipotong untuk pertama kalinya, lumpia terdiri dari 2 bagian sepanjang 500 cm dan 500 cm.
- Selanjutnya bagian kedua dari kiri dipotong, lumpia terdiri dari 3 bagian sepanjang 500 cm, 250 cm, dan 250 cm.

- Kemudian bagian kedua dari kiri dipotong, lumpia terdiri dari 4 bagian sepanjang 500 cm, 125 cm, 125 cm, dan 250 cm.
- Terakhir, bagian terkiri dipotong sehingga lumpia terdiri dari 5 bagian sepanjang 250 cm, 250 cm, 125 cm, 125 cm, dan 250 cm.

Jawaban:

1. C Version

```

1  #include <stdio.h>
2
3  int main() {
4      int N, L;
5      printf("Masukkan jumlah N: "); scanf("%d", &N);
6      printf("Masukkan panjang lumpia (L): "); scanf("%d", &L);
7
8      int B[N];
9      for (int i = 0; i < N; i++) {
10         printf("Masukkan urutan posisi potong sebanyak %d angka[%d/%d]: ", N, i+1, N);
11         scanf("%d", &B[i]);
12     }
13
14     // Array untuk menyimpan panjang bagian lumpia
15     int bagian[N + 1];
16     bagian[0] = L; // Bagian pertama adalah panjang lumpia awal
17
18     // Melakukan pemotongan
19     for (int i = 0; i < N; i++) {
20         int index = B[i] - 1; // Mengubah ke indeks 0
21         int panjang = bagian[index]; // Panjang bagian yang akan dipotong
22         bagian[index] = panjang / 2; // Setengah dari panjang bagian yang dipotong
23         // Memindahkan bagian yang dipotong ke bagian berikutnya
24         for (int j = N; j > index + 1; j--) {
25             bagian[j] = bagian[j - 1];
26         }
27         bagian[index + 1] = panjang / 2; // Menambahkan bagian baru
28     }
29
30     // Mencetak hasil
31     for (int i = 0; i <= N; i++) {
32         printf("%d ", bagian[i]);
33     }
34     printf("\n");
35
36     return 0;
37 }
38

```

2. C++ Version

```
1 #include <iostream>
2 #include <vector>
3 using namespace std;
4
5 void potong_lumpia(int n, int l, vector<int> &posisi_potong) {
6     vector<int> bagian = {1}; // Mulai dengan satu bagian
7
8     for (int i = 0; i < n; i++) {
9         int idx = posisi_potong[i] - 1; // Indeks bagian yang akan dipotong
10        int panjang_baru = bagian[idx] / 2;
11
12        // Sisipkan hasil potongan
13        bagian.insert(bagian.begin() + idx + 1, panjang_baru);
14        bagian[idx] = panjang_baru;
15    }
16
17    // Cetak hasil akhir
18    cout << "Hasil panjang bagian lumpia setelah dipotong: ";
19    for (int panjang : bagian) {
20        cout << panjang << " ";
21    }
22    cout << endl;
23 }
24
25 int main() {
26     int n, l;
27     cout << "Masukkan nilai N: "; cin >> n;
28     cout << "Masukkan panjang lumpia (L): "; cin >> l;
29
30     vector<int> posisi_potong(n);
31     for (int i = 0; i < n; i++) {
32         cout << "Masukkan urutan posisi potong sebanyak " << n << " angka: ";
33         cin >> posisi_potong[i];
34     }
35
36     potong_lumpia(n, l, posisi_potong);
37     return 0;
38 }
39
```