Nama : Abdillah Mufki Auzan Mubin

NPM : 40621100046

Tugas Algoritma dan Pemrograman I

1. Diketahui sebuah array tetap A={a, k, u, c,i, n, t, a, i, n, d, o, n, e, s, i, a}. Tentukan 3 nilai indeks terbesar yang nilai arraynya huruf konsonan menggunakan algoritma, Bahasa Pascal dan Bahasa C!

Solusi :

|  |  |  |
| --- | --- | --- |
| Bahasa Algoritmik | Bahasa Pascal | Bahasa C |
| judul: HurufKonsonan  Kamus Data:  A:array[0..16]of  Character=('a','k','u',  'c','i','n','t','a',  'i','n','d','o','n','e','s','i','a')i : Integer  Algoritma:  OUTPUT('3 Indeks terbesar berisi konsonan :')FORi = 0 TO16 DO  IF i ≥ 13 THEN  OUTPUT('A[',i,'] =',A[i])  ENDFOR | Program  HurufKonsonan;  Const  A : array[0..16] ofchar= ('a','k','u','c','i','n','t','a',  'i','n','d','o','n','e','s','i','a');  vari : integer;  begin  writeln('3 Indeks terbesar berisi konsonan : ');  fori := 0 to16 do  begin  ifi > 13 then  writeln('A[',i,'] = ',A[i]);  end; readln; end. | #include<stdio.h>  charA[]={'a','k','u',  'c','i','n','t','a',  'i','n','d','o','n','e','s','i','a'};  Int i;  main(){  printf("3 Indeks terbesar berisi konsonan :\n");  for(i = 0;i<=16;i++)  {  if(i > 13){  printf("A[%d] = %c\n",i ,A[i]);  }  }  } |

2.Diketahui sebuah array tetap A={a,l,g,o,r,i,t,m,a,p,e,m,r,o,g,r,a,m,a,n}. Hitunglah rata-rata nilai indeks array yang nilai array nya huruf konsonan menggunakan algoritma,Bahasa Pascal dan bahasa C !

Solusi :

|  |  |  |
| --- | --- | --- |
| Bahasa Algoritmik | Bahasa Pascal | Bahasa C |
| Judul: HitungNRRIndex  Kamus Data:  A: array[1..20] ofCharacter =('a','l','g','o','r','i','t','m',  'a','p','e','m','r','o','g'  ,'r','a','m','a','n')i, r,  sum: Integer  Algoritma:  sum ← 0FORi = 1 TO20 DO  sum ← sum+i  ENDFORr ← sum / 20  OUTPUT('rata-rata huruf konsonan : ', A[r]) | Program  HitungNRRIndex;  Const  A:array[1..20]ofchar=('a','l','g','o',  'r','i','t','m','a'  ,'p','e','m','r','o','g'  ,'r','a','m','a','n');  Var  i, r, sum : integer;  begin  sum := 0;  fori := 1 to20 do  begin  sum:=sum+i;end;  r := sum div20;  write('rata-rata huruf  konsonan : ', A[r]);  readln;  end. | #include<stdio.h>  charA[20] = {'a','l','g','o'  ,'r','i','t','m',  'a','p','e','m',  'r','o','g','r','a',  'm','a','n'};  inti, r, sum;  main()  {  sum = 0;  for(i=0;i<20;i++)  {  sum = sum+i;  }  r = sum / 20;  printf("rata-rata  huruf konsonan : %c",A[r]);  } |

3. Diketahui sebuah array tetap A={1,2,3,5,7,9,11,23,10,21,23,22,24,25,26,77}. Hitunglahrata-rata nilai array yang jika nilai indeksnya ditambahkan 5 akan habis dibagi 4menggunakan algoritma, Bahasa Pascal dan bahasa C !

Solusi :

|  |  |  |
| --- | --- | --- |
| Bahasa Algoritmik | Bahasa Pascal | Bahasa C |
| Judul: HitungNRR  Kamus Data:  A: array[1..16] ofinteger= (1,2,3,5,7,9,1  1,23,10,21,  23,22,24,25,26,77)  i , sum: Integer  avg : Integer  Algoritma :  Sum ← 0  For i = 1 to 16 DO  If I = 5 THEN  Sum ← sum+A[i]/4  ELSE  Sum ← sum+A[i]  ENDIF  ENDFOR  Avg ← sum/16  OUTPUR(avg) | Program  HitungNRR;  Const  A : array[1..16] ofinteger= (1,2,3,5,7,9,11  ,23,10,21,23,  22,24,25,26,77);  Var  i, sum : Integer;  avg : Integer;  begin  sum := 0;fori:=1 to16 do  begin  if i = 5 thensum :=sum+A[i] div4  else  sum := sum+A[i];  end;  avg := sum div16;  write(avg);  readln;  end. | #include<stdio.h>  Int A[16]={1,2,3,5,7,9,  11,23,10,21,23,  22,24,25,26,77  };  Int I, sum;  Int avg;  main()  {  Sum=0;  For (i=0; i<16; i++)  }  If (i==5){  Sum=sum+A[i]/4;  }else  {  Sum=sum+A[i];  }  }  avg =sum/16;  printf(%d”,avg);  } |

4. Diketahui suatu data tiket sbb.

|  |  |  |
| --- | --- | --- |
| Id\_ tiket | Jurusan | harga |
| T001  T002 | Bandung-Jakarta  Bandung-Bekasi | 85000  50000 |

Buatlah algoritma, bahasa Pascal dan bahasa C untuk memasukkan data tersebut terhadap struktur tiket menggunakan array! Program yang dibuat harus menggunakan 3 pilihan menu (1. Isi Data, 2. Tampilkan Data, 3. Keluar Program)

|  |  |  |
| --- | --- | --- |
| Bahasa Algoritmik | Bahasa Pascal | Bahasa C |
| Judul : DataBaseTiket  Kamus data :  Type datatiket :  Record<  Id : character  Harga : integer  >  DT : array [1..2] of datatiket  Menu, I : integer  Algoritma :  REPEAT  OUTPUT (‘1. Isi Data’)  OUTPUT (‘2. Tampilkan Data’)  OUTPUT (‘3. Keluar Program’)  OUTPUT (‘Masukan no.  menu :’)  INPUT(menu)  FOR I =1 to 2 do  Case (menu)  1:  OUTPUT (‘masukan id  tiket:’)  INPUT (DT [I] .Id)  OUTPUT (‘masukan jurusan :’)  INPUT (DT [I] .jurusan)  OUTPUT (‘masukan harga :’)  INPUT (DT [I] .harga)  2 :  OUTPUT(‘Id Tiket :  ‘, DT [i] .id)  OUTPUT(‘jurusan :  ‘, DT [i] .jurusan)  OUTPUT(‘Iharga :  ‘, DT [i] .harga)  3:  OUTPUT (‘Program Keluar!’)  ENDCASE  UNTIL Menu =3 | Program DataBaseTiket;  type datatiket=  record  id : string;  harga : integer;  end;  var  DT : array [1..2]of  datatiket;  menu, I :integer;  begin  repeat  writeln(‘1. Isi Data’)  writeln (‘2. Tampilkan Data’)  writeln (‘3. Keluar Program’)  Write (‘Masukan No. Menu :  ‘);  Readln(menu);  For I := 1 to 2 do  Case Menu of  1 :begin  Write(‘masukan ud tiket :’);  Readln(DT[i],id);  Write(‘masukan jurusan :’);  Readln(DT[i],jurusan);  Write(‘masukan harga :’);  Readln(DT[i],harga);  End;  2 : begin  Writeln(‘id tiket :’, DT[i].id);  Writeln(‘id jurusan :’, DT[i].jurusan);  Writeln(‘id harga :’, DT[i].harga);  End;  3: begin  Writeln (‘program keluar!’);  End;  End;  Until menu = 3;  End. | #include>stdio.h>  Typedef struct{  Char id[5];  Char jurusan [100];  Int harga;  }datatiket;  Datatiket DT[2];  Int menu, I;  Main(){  Do{  Printf(“1. Isi Data\n”);  Printf(“2. Tampilkan Data\n”);  Printf(“3.Keluar Program\n”);  Printf(“masukan NO.  Menu : “);  Scanf(“%d, &Menu);  For (i=0; i<=1; i++)  Scanf (%s”&DT[i].id);  fflush (stdin);  printf (“masukan jurusan : “);  scanf(“%s”,&DT[i]. jurusan);  fflush (stdin);  printf (“masukan harga : “);  scanf(“%s”,&DT[i]. harga);  }break;  Case 2 : { printf (“id tiket : %s\n”,DT[i].id);  printf (“jurusan : %s\n”,DT[i].jurusan);  printf (“harga : %s\n”,DT[i].harga);  }break;  Case 3 : {  Printf (“Program Keluar!\n”)  }break;  }  }while(menu !=3);  } |

5. Diketahui sebuah tiket untuk alat transportasi yang dipakai untuk mudik dengan harga tetap kemanapun sebagai berikut :

|  |  |  |
| --- | --- | --- |
| Kode | Jenis\_transportasi | Harga |
| 1 | Bus | 50.000 |
| 2 | Kereta Api | 100.000 |
| 3 | Ferry | 75.000 |
| 4 | Pesawat | 200.000 |

Jika calon penumpang membeli tiket lebih dari 3, maka berhak mendapat diskon sebesar 10% dari total pembelian. Buatlah program dengan bahasa C untuk memasukkan data rekappenjualantiketdenganstruktur: nama\_pemesan, jenis\_transportasi, total penjualanmenggunakanarrayberdasarkaninputan : nama\_pemesan, kode\_transportasi, danjumlah\_tiket. Kemudiantampilkankeseluruhanhasilstrukturrekappenjualan !

|  |  |
| --- | --- |
| INPUT | NamaPemesan : Rina  KodeTransportasi : 1  JumlahTiket : 2  Data BerhasilInput !Tambahlagi ?Ya NamaPemesan : Rini KodeTransportasi : 4 JumlahTiket : 3  Data BerhasilInput !Tambahlagi ?Tidak |
| OUTPUT | RekapPenjualanTiket :  1.  NamaPemesan : Rina  JenisTransportasi : Bus  Total Penjualan : 100000  2.  NamaPemesan : Rini  JenisTransportasi : Pesawat  Total Penjualan : 540000  ----------  Total : 640000 |

Solusi :

|  |  |
| --- | --- |
| Bahasa Algoritmik | Bahasa C |
| Judul : RekapPenjualanTIket  Kamus Data:  Type DataRekap : record<  Nama\_pemesan : character  Jumlah\_tiket : integer  Jenis\_transportasi : character  Harga : integer  Total\_penjualan : integer >  DR : array [0..10] of data rekap  I, j, diskon , total : integer  Tambah : character  Algoritma :  FOR I = 0 TO 10 DO  OUTPUT (‘Nama Pemesan : ’)  INPUT (DR[i].nama pemesan)  OUTPUT (‘Kode Transportasi :’)  INPUT (DR[i].kode)  CASE (DR[i].kode)  1:  DR[i].jenis\_transportasi ← ‘BUS’  DR[i].Harga ← 50000  2:  DR[i].jenis\_transportasi ← ‘Kereta Api’  DR[i].Harga ← 100000  3:  DR[i].jenis\_transportasi ← ‘ferry  DR[i].Harga ← 75000  4:  DR[i].jenis\_transportasi ← ‘pesawat’  DR[i].Harga ← 200000  ENDCASE  OUTPUT(‘Jumlah Tiket :’)  INPUT(DR[i].jumlahtiket)  IF(DR[i].jumlahtiket > 3)THEN  Diskon ← DR[i].Harga\*  DR[i].jumlahtiket\*10/100  DR[i].total\_penjualan ←  DR[i].Harga\*DR[i].jumlahtiekt – diskon | #include <stdio.h>  #include <stdlib.h>  #include <iostream>  #include <string>  using namespace std;  typedef struct{  char nama\_pemesan[100];  int jumlahtiket;  int kode;  string jenis\_transportasi;  int harga;  int total\_penjualan;  }DataRekap;  DataRekap DR [10];  Int I, j, diskon, total;  Char tambah;  Main()  {  For (i=0; i<=10; i++)  {  Printf (“nama pemesan : “);  Scaf (“%s”, &DR[i]. nama\_pemesan);  Printf(“kode Transportasi : “);  Scaf (“%d”, &DR[i].kode);  Switch(DR[i].kode)  {  Case 1 : {  DR[i].Jenis\_transportasi= “Bus”;  DR[i].Harga = 50000;  Break;  }  Case 2 : {  DR[i].Jenis\_transportasi=”Kereta Api”;  DR[i].Harga = 100000;  Break;  }  Case 3 : {  DR[i].Jenis\_transportasi=”Ferry”;  DR[i].Harga = 75000;  Break;  }  Case 4 : {  DR[i].Jenis\_transportasi=”Pesawat”;  DR[i].Harga = 200000;  Break;  }  }  Printf (“Jumlah Tiket : “);  Scanf (“%d”, &DR[i].Jumlahtiket);  If (DR[i].JumlahTiket >= 3)  {  diskon = DR[i].harga \*  DR[i]. JumlahTiket \* 10/100;  DR[i].total\_penjualan =  DR[i]Harga\*DR[i].jumlahtiket –  Diskon;  }else{  DR[i].total\_penjualan =  DR[i].Harga\*DR[i].jumlahtiket;  }  fflush(stdin);  printf(“Data Berhasil Input !  Tambah lagi ?”);  Scanf(“%c”, &tambah);  If(tambah==’N’)  {  J = I; i=10;  }  }  I=0;  System(“CLS”);  Printf(“Rekap penjualan Tiket :\n”);  While(I <= j )  {  Printf(“%d. \n”, DR [i].kode);  Printf(“nama pemean : %s\n”,  DR[i].nama\_pemesan);  Cout << “Jenis Transportasi :  “+DR[i].jenis\_transportasi << endl;  Printf(“total penjualan : %d\n”,  DR[i].total\_penjualan);  total =  total+DR[i].total\_penjualan;  i++;  }  Printf(“-------------------\n”);  Printf(“total : %d\n”, total);  } |

6. Buatlah algoritma untuk mengecek apakah sebuah kata yang diinputkan user termasuk polindrom atau tidak. Polindrom adalah kata yang jika dibalik urutan hurufnya, hasil bacanya tetap sama, contoh : katak, kasur rusak, dsb.

Contoh :

INPUT : Masukkan kata : katak

OUTPUT : Polindrom

INPUT : Masukkan kata : bisa

OUTPUT : Bukan Polindrom

|  |  |
| --- | --- |
| Bahasa Algoritmik | Bahasa C |
| Judul : Polindromkata  Kanus Data :  Kata : array[0..20]of char  I,pjg : integer  Kondisi :  Algoritma :  kondisi ← false  INPUT (kata)  Pjg ← strlen(kata)  FOR I = 0 TO pjg DO  IF (Kata[i] = kata [pjg -i-1]) THEN  Kondisi ← true  I ← pjg  ENDIF  ENDFOR  IF(kondisi)THEN  OUTPUT(‘Bukan Polindrom’)  ELSE  OUTPUT(‘polindrom’)  ENDIF | #include <iostream>  #include <mem.h>  Using namespace std;  Char kata[20];  Int I,pjg;  Bool kondisi;  main(){  kondisi = false;  cout << “masukan kata : “;  cin >> kata;  pjg = strlen(kata);  for (i=0; I<pjg; i++){  if (kata[i] != kata[pjg-i-1]){  kondisi = true;  I = pjg;  }  }  If (kondisi){  Cout << “Bukan Polindrom” << endl;  } else {  Cout << “Polindrom” << endl;  }  } |