**973)**

int k, ok, i;

int main()

{

char s[256], \*p;

cin.get(s,256);

p=strtok(s," ");

while(p)

{

ok=1;

for (i=0; i<strlen(p); i++)

if (!strchr("aeiou",p[i]))

{ok=0; break;}

if(ok) cout<<p<<endl;

p=strtok(NULL," ");

}

return 0;

}

**975)**

char s[256],\*p,sep[]=" ,.",mn[11];

int k;

int palindrom(char s[11])

{ int i,j;

i=0;

j=strlen(s)-1;

while(i<j)

{

if(p[i]!=p[j]) return 0;

i++;

j--;

}

return 1;

}

int main()

{

cin.getline(s,256);

p=strtok(s,sep);

while(p)

{

if(palindrom(p))

{ k++;

if (k==1) strcpy(mn,p); // inititializare cu primul palindrom

else if (strcmp(p,mn)<0) strcpy(mn,p);

}

p=strtok(NULL,sep);

}

if(k==0) cout<<"IMPOSIBIL";

else cout<<mn;

return 0;

}

**Obs**. Se poate initiliza mn cu cel mai mare sir de 10 litere mici

char s[256],\*p,sep[]=" ,.",**mn[11]="zzzzzzzzzz";**

if(palindrom(p))

{ k++;

if (strcmp(p,mn)<0) strcpy(mn,p);

}

**2797)**

**Sol 1**

int ok;

char s[101], mx[101], \*p;

int main()

{

cin.getline(s,101);

p=strtok(s," ,.");

while(p)

{

if(p[0]>='1'&& p[0]<='9')

{

ok=1;

if (p[0]>mx[0]) strcpy(mx,p);

}

p=strtok(NULL," ,.");

}

if(ok) cout<<mx;

else cout<<"nu exista";

return 0;

}

**Sol 2**

char s[101], mx[101], \*p;

int main()

{

cin.getline(s,101);

p=strtok(s," ,.");

while(p)

{

if(p[0]>='1'&& p[0]<='9' && p[0]>mx[0]) strcpy(mx,p);

p=strtok(NULL," ,.");

}

if(strlen(mx)>0) cout<<mx;

else cout<<"nu exista";

return 0;

}

**101)**

ifstream f("sortcuv.in");

ofstream g("sortcuv.out");

char s[251], v[251][21], \*p, aux[21];

int n, i, j;

int main()

{

f.get(s,251);

p=strtok(s, " ");

while(p)

{

strcpy(v[++n],p);

p=strtok(NULL, " ");

}

for(i=1;i<n;i++)

for(j=i+1;j<=n;j++)

if(strcmp(v[i],v[j])>0)

{ strcpy(aux,v[i]); // aux=v[i];

strcpy(v[i],v[j]); // v[i]=v[j];

strcpy(v[j],aux); // v[j]=aux;

}

for(i=1;i<=n;i++)

g<<v[i]<<endl;

return 0;

}

**2813)**

char s[101],\*p;

int i,ok;

int main()

{ cin.getline(s,101);

p=strtok(s," ");

while(p)

{

if(strstr(p,"DO") || strstr(p,"RE") || strstr(p,"MI") || strstr(p,"FA") || strstr(p,"LA") || strstr(p,"SI") || strstr(p,"SOL"))

{ cout<<p<<endl;

ok=1;

}

p=strtok(NULL," ");

}

if(!ok) cout<<"nu exista";

return 0;

}

**96)**

char s[256],\*p,sep[]=" ,.;:",r[21];

int palindrom(char p[])

{ int i,j;

char t[21];

strcpy(t,p);

for(i=0;i<strlen(t);i++)

if (t[i]>='A' && t[i]<='Z')

t[i]+=32;

i=0,j=strlen(t)-1;

while(i<j)

{

if(t[i]!=t[j]) return 0;

i++; j--;

}

return 1;

}

int main()

{

cin.getline(s,256);

p=strtok(s,sep);

while(p)

{

if(palindrom(p) && strlen(p)>strlen(r)) strcpy(r,p);

p=strtok(NULL,sep);

}

cout << r;

return 0;

}

**136)**

char s[151],\*p;

int k,i;

int main()

{

cin.get(s,151);

p=strtok(s," ");

while(p)

{

if(strchr("aeiouAEIOU",p[0]) && strchr("aeiouAEIOU",p[strlen(p)-1])) k++;

p=strtok(NULL, " ");

}

cout<<k;

return 0;

}

**799)**

char s[101],\*p, aux[101], t[101];

int main()

{

int k,ok=0;

cin.getline(s,101);

p=strtok(s," ");

while(p)

{

if(strlen(p)>=3 && strlen(p)%2==1)

{ k=strlen(p)/2;

strcpy(aux,p+k+1); strcpy(p+k,aux);

ok=1;

}

strcat(t,p); strcat(t," ");

p=strtok(NULL," ");

}

if(!ok) cout<<"nu exista";

else {

t[strlen(t)-1]=0; // sterg ultimul spatiu

s[0]=0; // sirul s devine sirul NULL

strcpy(s,t);

cout<<s;

}

return 0;

}

**2953)**

ifstream f("inserarechar.in");

ofstream g("inserarechar.out");

char t[500];

int ok;

int main()

{char s[256],c[16],\*p;

f.getline(c,16);

f.getline(s,256);

p=strtok(s," ");

while(p)

{ strcat(t,p);

if(strcmp(p,c)==0) {strcat(t,"?"); ok=1;}

strcat(t," ");

p=strtok(NULL," ");

}

if(ok) g<<t;

else g<<"NU APARE";

return 0;

}

**3112)**

char s[201],\*p;

int n,ok,x;

int main()

{cin.getline(s,201);

cin>>n;

p=strtok(s," ");

while(p)

{

if(strlen(p)==n) {cout<<p<<endl; ok=1;}

p=strtok(NULL," ");

}

if(!ok) cout<<"nu exista";

return 0;

}

**810)**

char s[71],\*p;

int k;

int main()

{ cin.getline(s,71);

p=strtok(s," ");

while(p!=NULL)

{ if(strchr(p,'a')) k++;

p=strtok(NULL," ");

}

cout<<k;

return 0;

}