

1.

6	0	
7	6	0
12		

2.

3.

```

int main()
{
    char *cumle;
    //Alınan cumle için rastgele bir kuantum belirlenir.
    cumle=(char*)malloc(100*sizeof(char));
    printf("Bir cumle giriniz:");
    gets(cumle);
    int kelime_syc=0,syc=0,enk=0,indis=0,i;
    //En uzun kelime kavutu degeri ilk kante kelime uzunlugu kadar olsun
    for(i=0; *(cumle+i)!='\0'; i++)
    {
        enk++;
    }
    for(i=0; *(cumle+i)!='\0'; i++)
    {
        if(*(cumle+i)==' ')
        {
            if(enk>syc)
            {
                enk=syc;
                indis=i-enk;
            }
            //Bir sonraki kelime için syc sıfırlanır.
            syc=0;
            kelime_syc++;
        }
        else
        {
            syc++;
        }
    }
    printf("\nKelime sayisi:%d\n", kelime_syc+1);
    printf("En kısa kelime uzunlugu:%d\n",enk);
    printf("En kısa kelime:");
    for(i=indis; i<indis+enk; i++)
    {
        printf("%c",*(cumle+i));
    }
    printf("\n");
    return 0;
}

```

```

TK *bilgi_al(int tSayisi)
{
    int i,j,kSayisi;
    TK *takimlar = (TK*)malloc(sizeof(TK)*tSayisi);
    for(i=0; i<tSayisi; i++)
    {
        printf("%d.Takim adi: ",i+1);
        scanf("%s", (takimlar+i)->ad);
        printf("%d.Takim turnuva sayisi: ",i+1);
        scanf("%d", &(takimlar+i)->turSayisi);
        //((takimlar+i)->macBilgiler = (MB*)malloc(sizeof(MB)*((takimlar+i)->turSayisi);
        (takimlar+i)->macBilgiler = (MB*)malloc(sizeof(MB)*((takimlar+i)->turSayisi);
        for(j=0; j<((takimlar+i)->turSayisi; j++)
        {
            ((takimlar+i)->macBilgiler + j)->toplamPuan =0;
            printf("%d.Takim için %d. turnuva tipi: ",i+1, j+1);
            scanf("%d", &((takimlar+i)->macBilgiler+j)->tTip);
            //scanf("%d", &aboneler[i].faturalar[i].fTip);
            printf("%d.Takim için %d. turnuvadaki galibiyet sayisi: ",i+1, j+1);
            scanf("%d", &((takimlar+i)->macBilgiler+j)->gal);
            printf("%d.Takim için %d. turnuvadaki beraberlik sayisi: ",i+1, j+1);
            scanf("%d", &((takimlar+i)->macBilgiler+j)->ber);
        }
    }
    return takimlar;
}

void puan_hesapla(TK *takimlar, int tSayisi)
{
    int i,j,turSayisi,tBorc;
    for(i=0; i<tSayisi; i++)
    {
        turSayisi = (takimlar+i)->turSayisi;
        tBorc = 0;
        for(j=0; j<turSayisi; j++)
        {
            ((takimlar+i)->macBilgiler + j)->toplamPuan =
            ((takimlar+i)->macBilgiler + j)->gal*3 + ((takimlar+i)->macBilgiler + j)->ber*1;
            printf("Takim Ad: %s, TurnuvaTip: %d, ToplamPuan: %d \n",
            (takimlar+i)->ad,((takimlar+i)->macBilgiler + j)->tTip,
            ((takimlar+i)->macBilgiler + j)->toplamPuan);
        }
        //((takimlar+i)->toplamPuan = tBorc;
    }
}

void lider_bul(TK *takimlar, int tSayisi)
{
    int i,j,t,turSayisi,tBorc;
    int lider_puan=0,lider_ind=0;

    for(t=1; t<4; t++)
    {
        lider_puan=0;
        lider_ind=0;
        for(i=0; i<tSayisi; i++)
        {
            turSayisi = (takimlar+i)->turSayisi;
            for(j=0; j<turSayisi; j++)
            {
                if(t==((takimlar+i)->macBilgiler + j)->tTip)
                {
                    if(((takimlar+i)->macBilgiler + j)->toplamPuan > lider_puan)
                    {
                        lider_ind=i;
                        lider_puan=((takimlar+i)->macBilgiler + j)->toplamPuan;
                    }
                }
            }
        }
        printf("TurnuvaTip: %d,lider Takim: %s, \n",t,(takimlar+lider_ind)->ad);
    }
}

```

4. a) Include, b) dinamik bellek yönetimi, c) Gösterici/Pointer, d) free , e) sizeof