EX 3 GREEDY APPROACH

1)

```
#include<stdio.h>
int main(){
    int arr[]={1,2,5,10,20,50,100,500,1000};
    int a;
    int count=0;
    int n=(sizeof(arr))/4;
    scanf("%d",&a);
    for(int i=n-1;i>=0;i--){
        int cc=a/arr[i];
        count+=cc;
        a=a%arr[i];
    }
    printf("%d",count);
}
```

2)

```
#include<stdio.h>
#include<stdlib.h>
int comp(const void *a,const void *b){
    return(*(int*)a-*(int*)b);
}
int main(){
    int m,n;
    scanf("%d",&m);
    int g[m];
    for(int i=0;i<m;i++){
        scanf("%d",&g[i]);
    }
    scanf("%d",&n);
    int s[n];
    for(int j=0; j<n;j++){
        scanf("%d",&s[j]);
    }
    qsort(g,m,sizeof(int),comp);
    qsort(s,n,sizeof(int),comp);
    int i=0,j=0,count=0;
    while(i<m && j<n){
        if(s[j]>=g[i]){
            count++;
            i++;
            j++;

        }
        else{
            j++;
        }
    }
    printf("%d\n",count);
    return 0;
}
```

3)

```
#include<stdio.h>
#include<stdlib.h>

int comp(const void* a, const void* b){
    return(*(int*)b-*(int*)a);
}
int main(){
    int m;
    scanf("%d",&m);
    int c[m];
    for (int i=0;i<m;i++){
        scanf("%d",&c[i]);
    }
    qsort(c,m,sizeof(int),comp);
    long long total=0;
    long long mul=1;
    for(int i=0;i<m;i++){
        total+=(long long)c[i]*mul;
        mul*=m;
    }
    printf("%lld\n",total);
    return 0;
}
```

4)

```
#include<stdio.h>
```

```
#include<stdlib.h>
int compare(const void *a, const void *b){
  return(*(int*)a- *(int*)b);
}

int maximizeSum(int arr[],int n){
    qsort(arr,n,sizeof(int),compare);
    int total=0;
    for(int i=0;i<n;i++){
       total+=arr[i]*i;
    }
    return total;
}
int main(){
   int n;
   scanf("%d",&n);

   int arr[n];
   for(int i=0;i<n;i++){
     scanf("%d",&arr[i]);
   }
   int result=maximizeSum(arr,n);
   printf("%d",result);
   return 0;
}
```

5)

```
#include <stdio.h>
#include <stdlib.h>


int compareAsc(const void* a, const void* b) {
   return (*(int*)a - *(int*)b);
}



int compareDesc(const void* a, const void* b) {
   return (*(int*)b - *(int*)a);
}

int main() {
   int N, i;
   scanf("%d", &N);
```

```
    int a[N], b[N];


    for(i = 0; i < N; i++) {
        scanf("%d", &a[i]);
    }


    for(i = 0; i < N; i++) {
        scanf("%d", &b[i]);
    }


    qsort(a, N, sizeof(int), compareAsc);


    qsort(b, N, sizeof(int), compareDesc);


    int sum = 0;
    for(i = 0; i < N; i++) {
        sum += a[i] * b[i];
    }

    printf("%d\n", sum);

    return 0;
}
```