Simulation of basic memory allocation and garbage collector

Program

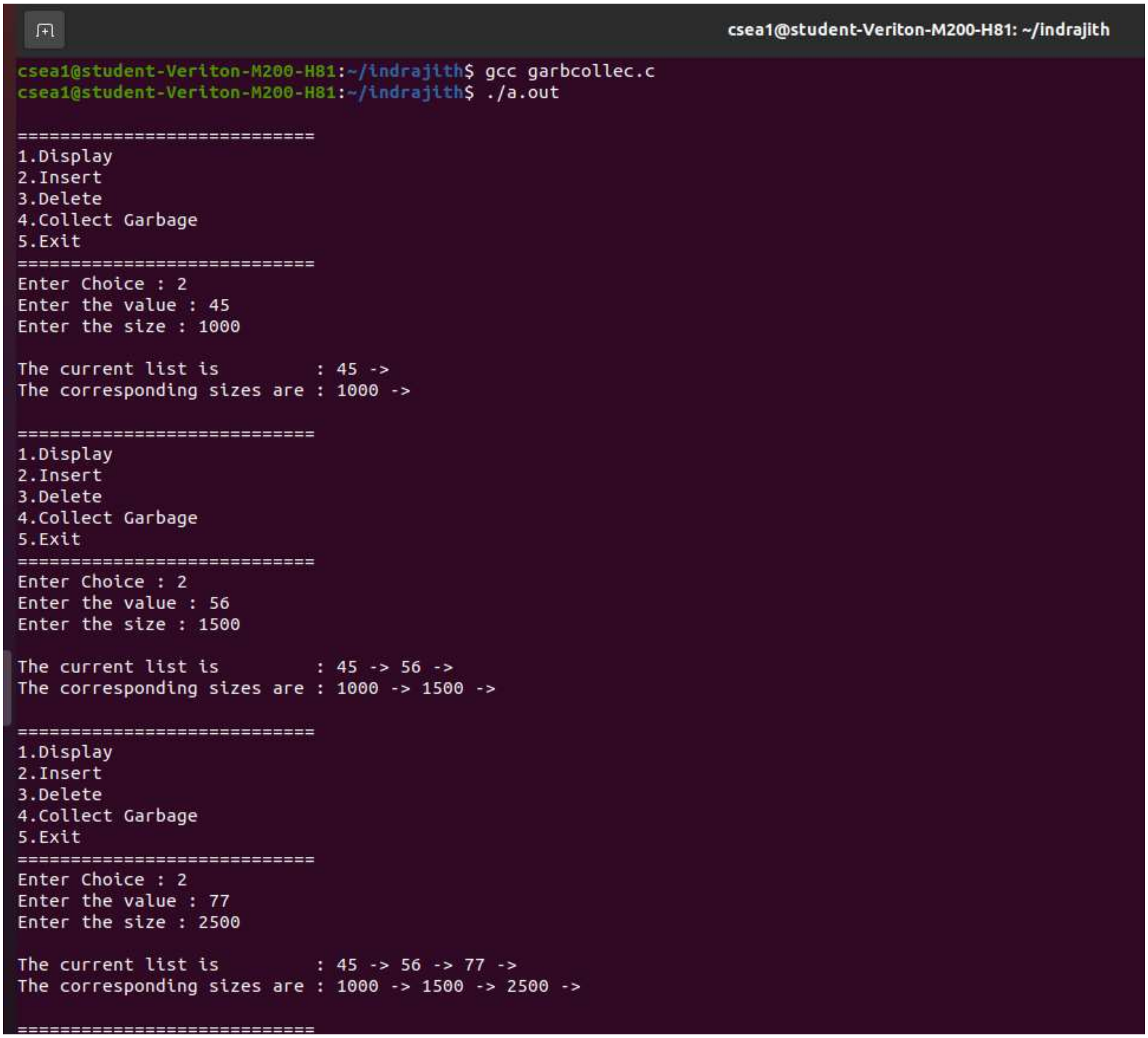
```
#include<stdlib.h>
#include<stdio.h>
#include<limits.h>
typedef struct node{
struct node* prev{
int value;
int size;
struct node* next;
}node;
node* create(){
node* t = malloc(sizeof(node));
printf("Enter the value : ");
scanf("%d",&(t->value));
printf("Enter the size : ");
scanf("%d",&(t->size));
t->next=NULL;
t->prev=NULL;
return t;
}
node* insert(node* t,node* H){ // inserting at front
if(H==NULL){
H=t;
}
else{
node* p = H;
while(p->next){
p=p->next;
}
p->next=t;
t->prev=p;
}
return H;
}
node* insertGarbage(int p,node* H){ // inserting garbage
node* t= malloc(sizeof(node));
t->next=NULL;
t->prev=NULL;
t->size=p;
t->value=INT_MIN;
if(H==NULL){
H=t;
```

```
}
else{
node* p = H;
while(p->next){
p=p->next;
}
p->next=t;
t->prev=p;
}
return H;
}
void display(node * H){
printf("\nThe current list is          : ");
node* p = H;
if(p){
while(p->next){
if(p->value!=INT_MIN){
printf("%d -> ",p->value);
}
else{
printf("# -> ");
}
p=p->next;
}
if(p->value!=INT_MIN){
printf("%d -> ",p->value);
}
else{
printf("# -> ");
}
}
printf("\nThe corresponding sizes are : ");
p = H;
if(p){
while(p->next){
printf("%d -> ",p->size);
p=p->next;
}
printf("%d -> ",p->size);
```

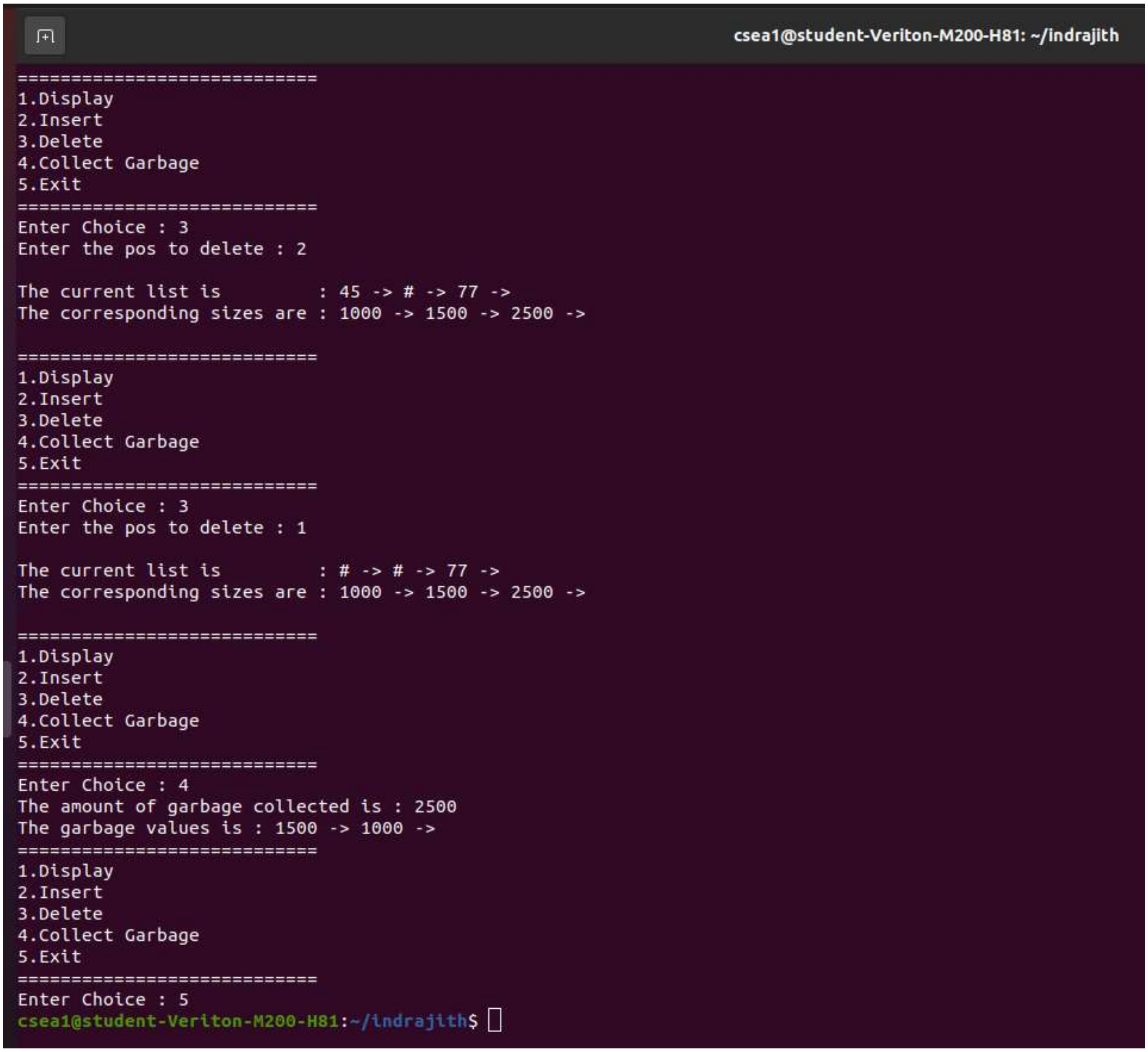
```
}
printf("\n");
}
void displayGarbage(node * HG){
printf("\nThe garbage values is : ");
node* p = HG;
if(p){
while(p->next){
printf("%d -> ",p->size);
p=p->next;
}
printf("%d -> ",p->size);
}
}
node* delete(node* H,node* HG){
int pos;
printf("Enter the pos to delete : ");
scanf("%d",&pos);
node* p = H;
for(int i =2;i<=pos;i++){
p=p->next;
}
if(p->value!=INT_MIN){
p->value=INT_MIN;
HG= insertGarbage(p->size,HG);
}
display(H);
return HG;
}
void garbageCollection(node* H,node* HG){
node * p =H;
int garbage = 0;
if(p){
while(p->next){
if(p->value==INT_MIN){
garbage += p->size;
}
p=p->next;
}
if(p->value==INT_MIN){
garbage += p->size;
}
}
printf("The amount of garbage collected is : %d",garbage);
```

```
displayGarbage(HG);
}
int main(){
node* HL = NULL;
node* HG = NULL;
int n,pos;
while (1){
printf("\n============================");
printf("\n1.Display\n2.Insert\n3.Delete\n4.Collect Garbage\n5.Exit");
printf("\n============================");
printf("\nEnter Choice : ");
fflush(stdin);
scanf("%d", &n);
node* t;
switch (n) {
case 1:
display(HL);
break;
case 2:
t=create();
HL=insert(t,HL);
display(HL);
break;
case 3:
HG = delete(HL,HG);
break;
case 4:
garbageCollection(HL,HG);
break;
case 5:
exit(0);
default :
printf("Wrong option selected");
}
}
}
```

Output



```
csea1@student-Veriton-M200-H81:~/indrajith$ gcc garbcollec.c
csea1@student-Veriton-M200-H81:~/indrajith$ ./a.out

============================
1.Display
2.Insert
3.Delete
4.Collect Garbage
5.Exit
============================
Enter Choice : 2
Enter the value : 45
Enter the size : 1000

The current list is         : 45 ->
The corresponding sizes are : 1000 ->

============================
1.Display
2.Insert
3.Delete
4.Collect Garbage
5.Exit
============================
Enter Choice : 2
Enter the value : 56
Enter the size : 1500

The current list is         : 45 -> 56 ->
The corresponding sizes are : 1000 -> 1500 ->

============================
1.Display
2.Insert
3.Delete
4.Collect Garbage
5.Exit
============================
Enter Choice : 2
Enter the value : 77
Enter the size : 2500

The current list is         : 45 -> 56 -> 77 ->
The corresponding sizes are : 1000 -> 1500 -> 2500 ->

============================
```



```
============================
1.Display
2.Insert
3.Delete
4.Collect Garbage
5.Exit
============================
Enter Choice : 3
Enter the pos to delete : 2

The current list is         : 45 -> # -> 77 ->
The corresponding sizes are : 1000 -> 1500 -> 2500 ->

============================
1.Display
2.Insert
3.Delete
4.Collect Garbage
5.Exit
============================
Enter Choice : 3
Enter the pos to delete : 1

The current list is         : # -> # -> 77 ->
The corresponding sizes are : 1000 -> 1500 -> 2500 ->

============================
1.Display
2.Insert
3.Delete
4.Collect Garbage
5.Exit
============================
Enter Choice : 4
The amount of garbage collected is : 2500
The garbage values is : 1500 -> 1000 ->
============================
1.Display
2.Insert
3.Delete
4.Collect Garbage
5.Exit
============================
Enter Choice : 5
csea1@student-Veriton-M200-H81:~/indrajith$ 
```