**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **17/06/2020** | | | | | **Name:** | **Shwetha** | |
| **Sem & Sec** | **8th B** | | | | | **USN:** | **4AL16CS101** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **-** | | | | | | |
| **Max. Marks** | | **-** | | **Score** | | | **-** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **Artificial intelligence in python** | | | | | | | |
| **Certificate Provider** | | | **Great learning** | | **Duration** | | | **7hr** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:**  **C program for triply linkedlist.** | | | | | | | | |
| **Status: Solved**  **Solution link: https://github.com/alvas-education-foundation/Shwetha-** | | | | | | | | |
| **Uploaded the report in Github** | | | | | **Yes** | | | |
| **If yes Repository name** | | | | | **Shwetha-** | | | |
| **Uploaded the report in slack** | | | | | **Yes** | | | |

Certification Course Details:



*Coding Challenges Details:*

*#include<stdio.h>*

*structSLL;*

*structTLL{*

*structTLL\*top;*

*structTLL\*bottom;*

*structSLL\*next;*

*};*

*typedefstructTLLtnode;*

*typedefstructSLL{*

*charch;*

*structSLL\*link;*

*};*

*typedefstructSLLsnode;*

*snode\*newnode,\*ptr,\*prev,\*temp;*

*snode\*first=NULL,\*last=NULL;*

*tnode\*newt,\*tlast=NULL,\*ttemp;*

*//---TLLnode---*

*tnode\*create\_tnode()*

*{*

*newt=(tnode\*)malloc(sizeof(tnode));*

*if(newt==NULL)*

*{*

*printf("\nMemorywasnotallocated");*

*return0;*

*}*

*else*

*{*

*newt->top=NULL;*

*newt->bottom=NULL;*

*newt->next=NULL;*

*returnnewt;*

*}*

*}*

*//---SLL---*

*snode\*create\_node(charc)*

*{*

*newnode=(snode\*)malloc(sizeof(snode));*

*if(newnode==NULL)*

*{*

*printf("\nMemorywasnotallocated");*

*return0;*

*}*

*else*

*{*

*newnode->ch=c;*

*newnode->link=NULL;*

*returnnewnode;*

*}*

*}*

*//---insertSLL---*

*voidinsert\_node\_first(charc)*

*{*

*newnode=create\_node(c);*

*if(tlast->next==NULL)*

*tlast->next=newnode;*

*if(first==last&&first==NULL)*

*{*

*first=last=newnode;*

*first->link=NULL;*

*last->link=NULL;*

*}*

*else*

*{*

*temp=first;*

*first=newnode;*

*first->link=temp;*

*}*

*printf("\n----INSERTED%cTOSLL----",c);*

*}*

*//---insertTLL---*

*voidinsert\_Tnode()*

*{*

*newt=create\_tnode();*

*if(tlast==NULL)*

*{*

*tlast=newt;*

*tlast->next=NULL;*

*tlast->top=NULL;*

*tlast->bottom=NULL;*

*}*

*else*

*{*

*ttemp=tlast;*

*tlast=newt;*

*tlast->next=NULL;*

*tlast->top=ttemp;*

*tlast->bottom=NULL;*

*ttemp->bottom=tlast;*

*}*

*printf("\n----CREATEDNEWTLL----");*

*}*

*voidmain()*

*{*

*chars[100],n;*

*inti;*

*scanf("%[^;]s",s);*

*insert\_Tnode();*

*for(i=0;s[i]!='\0';i++)*

*{*

*n=s[i];*

*if(n=='\n')*

*insert\_Tnode();*

*else*

*insert\_node\_first(n);*

*}*

*printf("\n%s\n",s);*

*}*