

1 întrebare

Nu a primit
răspuns încă

Marcat din 1,00

Care conditie trebuie folosita in parcurgerea unei liste pentru inserarea in lista simplu inlantuitata, astfel incat lista sa fie sortata crescator?

Alegeți o opțiune:

- ☐ a. `while(p->next && p->info<infoDeCautat);`
- ☐ b. `while(p->next && p->next->info<infoDeCautat);`
- ☒ c. `while(p && p->info<infoDeCautat);`
- ☐ d. `while(p->next || p->next->info>infoDeCautat);`

2 întrebare

Nu a primit
răspuns încă

Marcat din 1,00

O structura de tip stiva nu poate fi reprezentata prin intermediul:

Alegeți o opțiune:

- ☒ a. unui arbore binar
- ☐ b. unei liste dublu inlantuita
- ☐ c. unei liste simplu inlantuita
- ☐ d. unui vector

3 întrebare

Nu a primit
răspuns încă

Marcat din 1,00

Se considera structura `struct Exemplu{ int id; char* nume}`. Se declara pointerul Exemplu * ptrEx. Care instructiune este incorecta?

Alegeți o opțiune:

- ☐ a. `ptrEx->nume=(char*)malloc(sizeof(char)*(strlen("ex")+1));`
- ☐ b. `(*ptrEx).nume=(char*)malloc(sizeof(char)*(strlen("ex")+1));`
- ☒ c. `ptrEx.id=15;`
- ☐ d. `ptrEx->id=10;`

4 întrebare

Nu a primit
răspuns încă

Marcat din 1,00

Care din următoarele variante nu reprezintă o metoda de evitare a coliziunilor într-o tabela de dispersie:

Alegeți o opțiune:

- ☒ a. overflow area
- ☐ b. re-hashing
- ☐ c. sharing
- ☐ d. chaining;

5 întrebare

Nu a primit
răspuns încă

Marcat din 1,00

Ce efect are operatorul ++ asupra unui pointer de tip Element*?

Alegeți o opțiune:

- ☐ a. deplasarea pointerului înainte cu dimensiunea primului camp din structura Element, exprimata in octeti;
- ☒ b. deplasarea pointerului înainte cu dimensiunea structurii Element, exprimata in octeti;
- ☐ c. deplasarea pointerului inapoi cu 1 octet;
- ☐ d. deplasarea pointerului înainte cu 1 octet;

6 întrebare

Nu a primit
răspuns încă

Marcat din 1,00

Care instructiune realizeaza afisarea tuturor elementelor, de tip int, dintr-o lista simplu inlantuita, identificata prin p-adresa primului nod?

Alegeți o opțiune:

- ☐ a. `while(p){printf("%d",p->info); p=p->info;};`
- ☐ b. `while(p->next){printf("%d",p->info); p=p->next;};`
- ☒ c. `while(p){printf("%d",p->info); p=p->next;};`
- ☐ d. `while(p->next){printf("%d",p->info); p=p->next;};`

7 întrebare

Nu a primit
răspuns încă

Marcat din 1,00

Ce instrucțiuni se folosesc pentru stergerea primului nod dintr-o lista simplu înlantuită, în care informația utilă este `char*nume`?

Alegeți o opțiune:

- ☐ a. `if(cap){nod*aux=cap->next; free(cap->nume); free(cap); aux=cap;}`
- ☒ b. `if(cap){nod*aux=cap->next; free(cap->nume); free(cap); cap=aux;}`
- ☐ c. `if(cap){nod*aux=cap->next; free(cap); free(cap->nume); cap=aux;}`
- ☐ d. `if(cap){nod*aux=cap->info; free(cap->nume); free(cap); cap=aux;}`

8 întrebare

Nu a primit
răspuns încă

Marcat din 1,00

Care dintre următoarele diferite dintre un vector și o listă simplu înlanțuită nu este adevărată?

Alegeți o opțiune:

- ☐ a. elementele vectorului pot fi accesate direct, iar elementele listei sunt accesate secvențial
- ☒ b. memoria pentru listă este alocată doar la execuție iar memoria pentru vector este alocată doar la compilare
- ☐ c. pentru listă nu trebuie cunoscut a priori numărul de elemente, iar la vector trebuie cunoscut numărul de elemente înainte de a se alocă spațiu.
- ☐ d. vectorul are zonă de memorie alocată contiguu, iar memoria ocupată de nodurile listei nu este contiguă

9 întrebare

Nu a primit
răspuns încă

Marcat din 1,00

O lista simplu inlantuita este referita prin:

Alegeți o opțiune:

- ☒ a. adresa primului nod al listei;
- ☐ b. primul element din lista;
- ☐ c. adresa unui nod din cadrul listei;
- ☐ d. variabila cap de tip nod*;

10 Întrebare

Nu a primit
răspuns încă

Marcat din 1,00

Care instrucțiune de citire a variabilei char aux[20] este corectă?

Alegeți o opțiune:

- ☐ a. scanf("%d",&aux);
- ☐ b. scanf("&s",&aux);
- ☒ c. scanf("%s",&aux);
- ☐ d. scanf("&d",%aux);

11 Întrebare

Nu a primit
răspuns încă

Marcat din 1,00

Care este definirea corecta a structurii unui nod dintr-o lista dublu inlantuita cu informatia utila de tip char*?

Alegeți o opțiune:

- ☐ a. struct* nod{ char** info; nod* next; nod* prev;};
- ☒ b. struct nod{ char* info; nod* next; nod* prev;};
- ☐ c. struct nod{ char* info; nod* next; nod* next;};
- ☐ d. struct nod{ char info; nod next; nod prev;};

12 Întrebare

Nu a primit
răspuns încă

Marcat din 1,00

Care este instructiunea, care alocă în mod corect spațiu pentru memorarea cuvântului "SDD"?

Alegeți o opțiune:

- ☐ a. `char* variabila=malloc(sizeof(char)*(strlen("SDD")+1));`
- ☐ b. `char* variabila=(char*)malloc(sizeof(char)*(strlen("SDD")+1));`
- ☒ c. `char*variabila=(char*)malloc(sizeof(char)*(strlen("SDD")+1));`
- ☐ d. `char* variabila=(char)malloc(sizeof(char)*(strlen("SDD")));`

