هياكل بيانات وخوارزميات
9:11
السبت 19/6/2021
د/مصطفى أبوبكر عبدالمجيد سالم

Faculty of Computers & Information, Assiut University
2nd Level Bioinformatics program (Group 3& 4 Only)
Final Exam
Duration: 2 hours

1

* الإسم الرباعي (بالعربي فقط)

ماريا سامح الفونس قزمان

2

* رقم الجلوس

1620195209

3

المستوي *

- الاول
- الثاني
- الثالث
- رابعة 2013
- رابعة 2014
- رابعة 2015
- رابعة 2016
- رابعة 2017

4

البرنامج *

- عام
- بايو
- هندسة

5

رقم المعمل *

- ج.
- د.

- اب
- اد
- اهـ
- أ٢
- ب٢
- ج٢
- د٢
- هـ٢
- أ٣
- ب٣
- ج٣
- د٣
- هـ٣
- أ٤
- ب٤

6

* رقم الكمبيوتر

19

7

* الكود (قد تمت مراجعة بيانات الطالب ورقم الجلوس)

rRPc

8

what is the output of the following code?
(2 Points)

```
int arr[4][2] = {
    {10, 20},
    {30, 40},
    {50, 60},
    {70, 80}
     };

     for (int i = 2; i < 4; i++)
         for (int j = 0; j < 2; j-
             cout << arr[i][j] <<
```

- 50 60 70 80
- 50 70 60 80
- 10 30 20 40
- 10 20 30 40

9

The array for the dynamically allocated array-based sorted list is allocated in what part of storage?
(2 Points)

- The heap
- The stack
- The cloud
- It depends on the compiler.

10

The member variables and functions declared following the word "_____" are accessible to the client program.
(2 Points)

- public
- private
- None of the above

11

The assertion that states what is true before execution of a code segment is known as_____
(2 Points)

- requirement.
- loop invariant.
- postcondition.

- precondition.

12

When a class member function is a binary operation, how are the operands specified?
(2 Points)

- Both operands are passed to the member function as parameters.
- One operand is passed to the member function and the second is a global variable.
- One operand is passed to the member function and the second is the class instance to which the function is applied.
- It depends on the compiler.

13

The logical view of a data structure is associated with which of the following?
(2 Points)

- What?
- Why?
- How?
- None of the above

14

const int* x;
Choose the right option for x?
(2 Points)

- The pointer x can be modified to point to any appropriate data item

- Data can be modified through the dereferenced pointer
- Pointer always points to the same memory location, and the data at that location cannot be modified via the pointer.
- None of the above

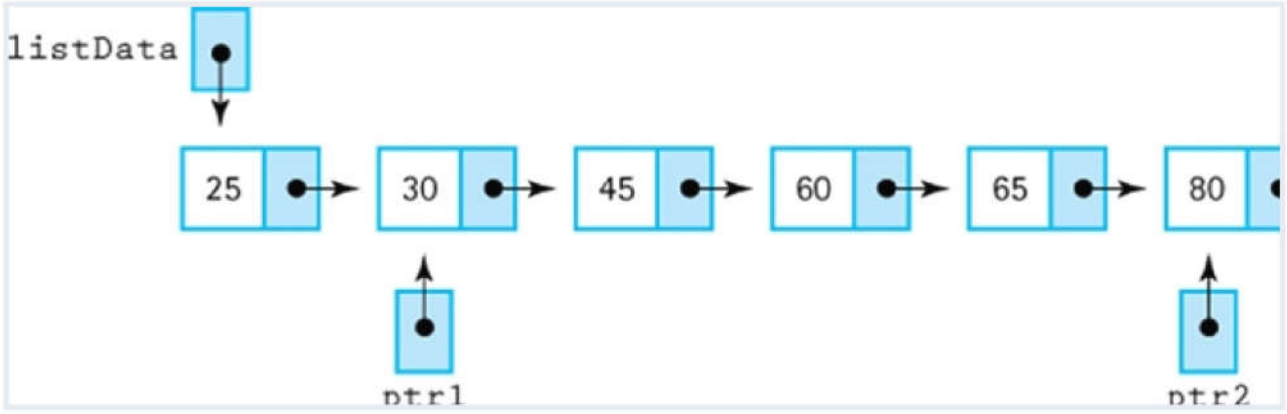
15

The basic operations that change the state of one or more of the data values.
(2 Points)

- Iterators
- Transformers
- Accessor
- Predicates

16

Use the linked list pictured below.
What is the correct code to make ptr2 points to the last node in the list?
(2 Points)



- ptr2 = ptr1->next
- ptr2 = ptr2->next
- ptr2 = NULL

- None of the above

17

In linked representation of stack, the null pointer of the last node in the list signals _____.
(2 Points)

- Beginning of the stack
- Middle of the stack
- Bottom of the stack
- In between some value

18

What are the disadvantages of arrays?
(2 Points)

- Data structure like queue or stack cannot be implemented
- There are chances of wastage of memory space if elements inserted in an array are lesser than the allocated size
- Index value of an array can be negative
- Elements are sequentially accessed

19

When writing the code to define a member function, the name of ____ must precede the function name with the ____ operator in between.
(2 Points)

- the class, ::

- the record, ::
- the data member, ::
- the class, . (dot)

20

Is this function prototype correct: void print2DArray(const int a[][]);
(2 Points)

- True
- False

21

If we compare array and linked list then which of the following point is not true about the linked list?
(2 Points)

- It is easy to delete elements in Linked List
- Access of elements in linked list takes less time than compared to arrays
- Linked–lists are dynamic in nature
- In linked list, random access is not allowed

22

To implement a list ADT for which the number of components is unknown and can vary widely, which of the following is the best choice?
(2 Points)

- A linked list represented as a one-dimensional array of structs

- A two-dimensional array of the component type
- A one-dimensional array of the component type
- A linked list represented as dynamic structs and pointers

23

The name of array stores the _____ of the first array element.
(2 Points)

- memory address
- value
- element number
- data type

24

With a list ADT, insertions and deletions at the front of the list are slower with a linked list representation than with a direct array representation.
(2 Points)

- True
- False

25

The elements in an array are logically homogeneous when they have the same data type.
(2 Points)

- True

False

26

_____ is a way of modeling real-life data in a specific context; also called the problem domain.
(2 Points)

Logical level

Application level

Implementation level

None of the above

27

What is special about the last node in a dynamic linked list?
(2 Points)

Its component (data) member is empty

Its link member (next) contains the value NULL

It has no link member

Its component (data) member contains the value 0

28

The next item in a linked list always can be found by accessing the next physical location in memory.
(2 Points)

True

False

29

```
int *secret;
secret = new int[10];
secret[0] = 10;
for (int j = 1; j < 10; j++)
secret[j] = secret[j -1] + 5;
for(int j = 0; j < 10; j++)
cout << secret[j] << " ";
cout << endl;
```

What is the output of the following code?
(2 Points)

- 10 15 20 25 30 35 40 45 50
- 15 20 25 30 35 40 45 50 55 60
- 10 15 20 25 30 35 40 45 50 55
- The code has errors.

30

A pointer can be initialized with
(2 Points)

- Null
- Zero
- Address of an object of same type
- All of the above

31

Suppose that a C++ class D is derived from a base class B. Class B has a public member function Func() that is declared to be virtual, and class D redefines its own version of Func(). At execution time, suppose that a D object is passed to the following function:

```
void DoSomething( B& x )
{
    x.Func();
}
```

Within the DoSomething function, whose version of Func() is called?
(2 Points)

- B's version
- D's version
- Both B's version and D's version
- None of these answers is correct.

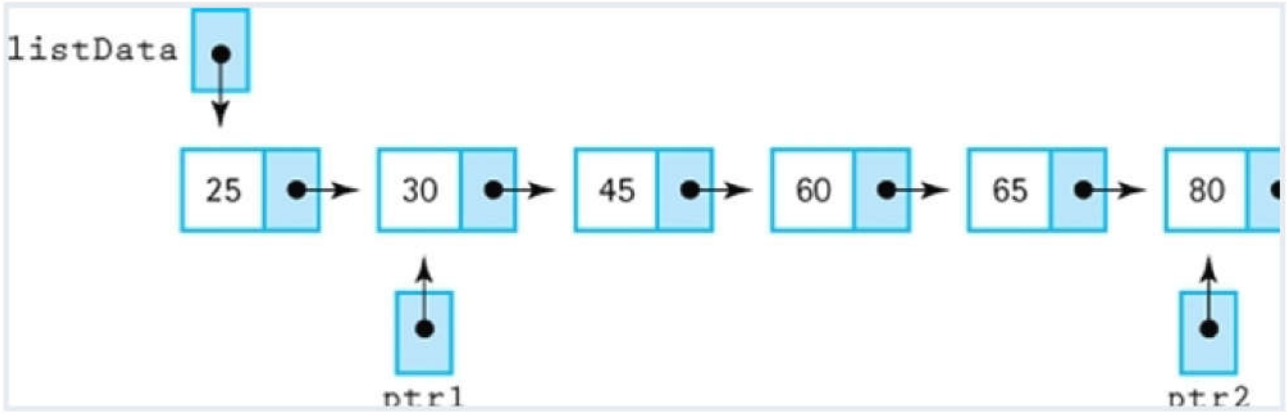
32

In C++, if class X is a base class of class Y, then Y cannot directly access X's private data.
(2 Points)

- True
- False

33

Use the linked list pictured below.
ptr2->next->next = _____ while listData->next->next->info = _____
(2 Points)



- 90 , 45
- 80, 45
- NULL, 45
- None of the above

34

Which of the following correctly fills the blank in the line labeled "// 1" and "// 2"?
(2 Points)

```
void StackType::Push(ItemType newItem)
// Adds newItem to the top of the stack.
// Pre:  Stack has been initialized.
// Post: If stack is full, FullStack exception is
//       else, newItem is at the top of the stack
//topPtr points to the top of the stack
{
  if (IsFull())
    throw FullStack();
  else
  {
    NodeType* location;
    location = new NodeType;
    location->info = newItem;
      _____________________;   //1
      _____________________;   //2

  }
}
```

- 1) location->next=topPtr; 2)topPtr=location->next
- 1) location->next=topPtr; 2) topPtr=location
- 1) location->next=topPtr; 2) topPtr++
- 1) topPtr=location 2) location->next=topPtr->next;

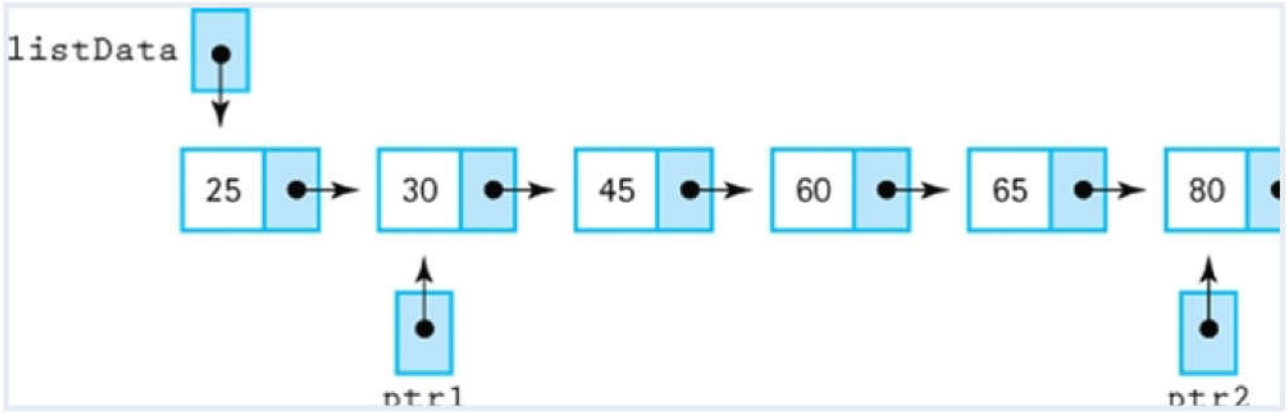
35

An abstraction is a model of a complex system that includes only the essential details.
(2 Points)

- True
- False

36

Use the linked list pictured below. What is the correct code to set the info member of the node containing 45 to 100?
(2 Points)



- ptr2->next->info = 100
- listData->next->info = 100
- ptr1->next->next->info = 100
- ptr1->next->info = 100

37

Function prototypes are_____
(2 Points)

- declarations
- definitions
- Bother of these answers are correct
- None of the above is correct

38

Which of the following correctly fills the blank in the line labeled "// 1" and "// 2"?
"DeleteItem for Unsorted array-based list"
(2 Points)

```
void UnsortedType::DeleteItem(ItemType item)
// Pre:  item's key has been initialized.
//       An element in the list has a key that matche
// Post: No element in the list has a key that matche
{
  int location = 0;

  while (item.ComparedTo(info[location]) != EQUAL)
    _____________________;  //1

 _____________________;  //2

 length--;
}
```

- 1) location++; 2) info[location] = info[length - 1];
- 1) info[location] = info[length - 1]; 2) location++;
- 1) location++; 2) info[location] = info[length];
- 1) location--; 2) info[location] = info[length];

39

The elements in an array are physically homogeneous.
(2 Points)

- True
- False

40

If currPtr points to a node in a dynamic linked list, the operation currPtr++ advances to the next node in the list.
(2 Points)

- True
- False

41

Is this function prototype correct: void print1DArray(const int a[]);
(2 Points)

- True
- False

42

What is the output of the following code?
(2 Points)

```
int x;
int *p;
int *q;
p = new int[10] ;
q = p;
*p = 4;
for(int j = 0; j < 10; j++)
{
        x = *p;
        p++;
        *p = x + j;
}
for (int k = 0; k < 10; k++)
{
        cout << *q << " ";
        q++;
}
cout << endl;
```

- 4 5 6 7 8 9 10 11 12 13
- 4 4 5 7 10 14 19 25 32 40
- 4 5 7 10 14 19 25 32 40 50
- The code has errors

43

Which of the following operations are dependent on the length of the linked list if we have pointers to the first and the last node of a singly linked list?
(2 Points)

- Delete the first element
- Delete the last element of the list
- Insert a new element as a first element
- Add a new element at the end of the list

44

string* x, y; Choose the right option?
(2 Points)

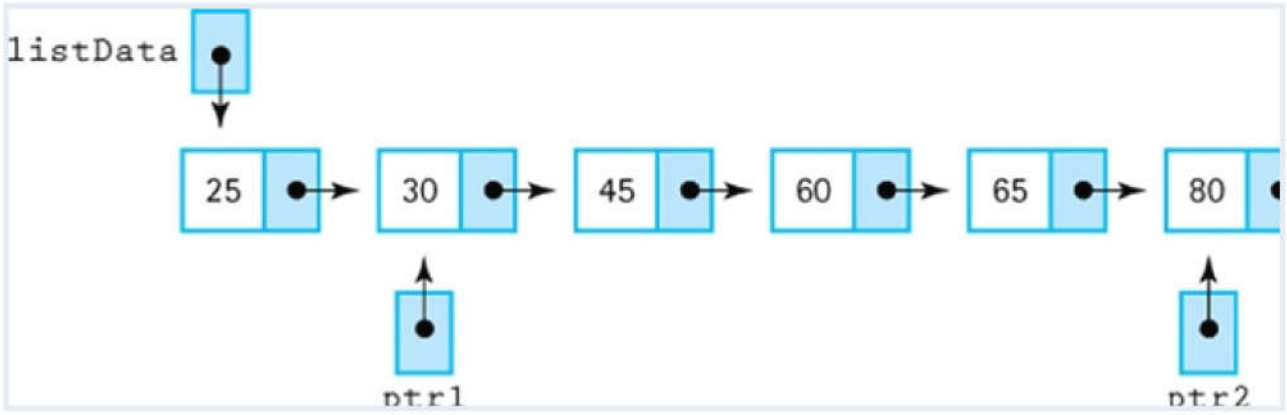
- x is a pointer to a string, y is a string
- y is a pointer to a string, x is a string
- Both x and y are pointers to string types
- None of the above

45

Use the linked list pictured below, what is the output of the following code (info: data, next: pointer to the next node)?

```
int s=0;
currPtr = listData->next:
while(currPtr->next != NULL)
{
  currPtr = currPtr->next;
  s+= currPtr->info;
}
cout << s<
```

(2 Points)



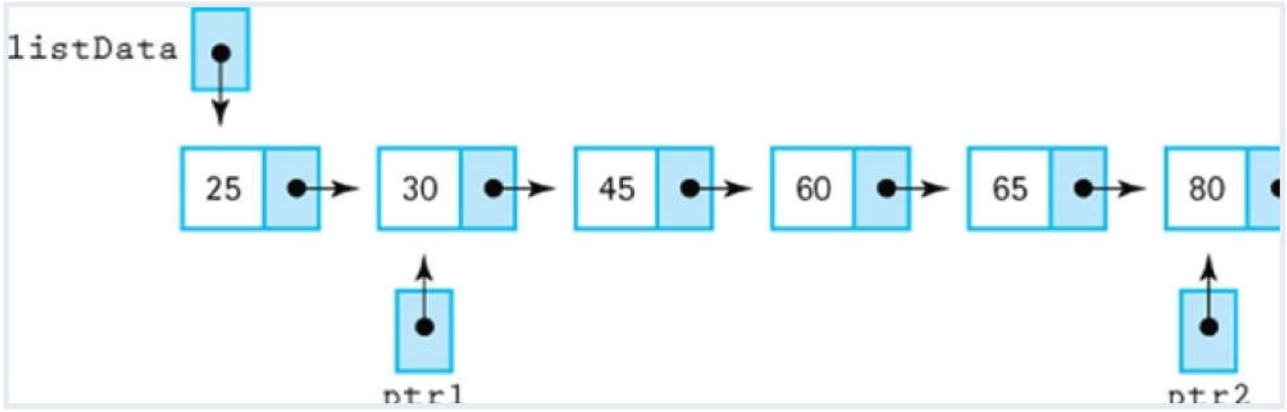
- 370
- 340
- 395
- 250

46

Use the linked list pictured below.

ptr1->next->info == 60 ?

(2 Points)



True

False

47

Which of the following statements in a client program correctly prints out the year of the variable day1 of type DateType?
(2 Points)

```
class DateType
{
public:
   void Initialize(int, int, int);
   int GetYear() const;              // returns yea
   int GetMonth() const;             // returns mor
   int GetDay() const;               // returns day

private:
   int  year;
   int  month;
   int  day;
};
```

- cout << day1.GetYear;
- cout << day1.GetYear();
- The year cannot be printed by a client program
- cout << GetYear.day1;

48

Read the following code and choose the correct choice?

```
int *p;                                  // Line 1
int *q;                                  // Line 2
p = new int;                             // Line 3
*p = 43;                                 // Line 4
q = p;                                   // Line 5
*q = 52;                                 // Line 6
delete q;                                // Line 7
cout << *p << " " << *q << endl;         // Line 8
```

(2 Points)

- The statement in Line 5 copies the value of p into q
- Both p and q point to the same memory location.
- The statement in Line 7 deallocates the memory space pointed to by q, which in turn invalidates both p and q.
- The values printed by the statement in Line 8 are unpredictable.
- All the above are correct

49

The file containing the definitions of the member functions of class DateType is called the _____ file.
(2 Points)

- specification

- implementation
- client
- None of the above

50

Deleting from an unsorted list requires that the elements below the one being deleted be moved up one slot.
(2 Points)

- True
- False

51

What is the goal of information hiding?
(2 Points)

- Cutting down on gossip.
- Controlling access between functions.
- Controlling access to the details of a function.
- None of the above.

52

If we want to convert infix notation (e.g. + 2 5) to postfix notation (e.g. 2 5 +), then which of the following data structure should we use?
(2 Points)

- HashMap

- Queue
- Singly Linked List
- Stack

53

Which of the following correctly fills the blank in the line labeled "// 1" and "// 2"?
(2 Points)

```
ItemType SortedType::GetItem(ItemType& item, bool& found)
                    // Uses binary search algorithm
                    {
                      int midPoint;
                      int first = 0;
                      int last =length - 1;
                      bool moreToSearch = first <= last;
                      found = false;
                      while (moreToSearch && !found)
                      {
                        midPoint = (first + last) / 2;
                        switch (item.ComparedTo(info[midPoint]))
                        {
                            case LESS    : last = _____________;
                                           moreToSearch = first <= last;
                                           break;
                            case GREATER : first = _____________;
                                              moreToSearch = first <=
                                           break;
                            case EQUAL   : found = true;
                                           item = info [midpoint];
                                           break;
                        }
                      }
                     return item;
                    }
```

- [x] 1) midPoint -1; 2) midPoint + 1;
- [ ] 1) midPoint +1; 2) midPoint + 1;
- [ ] 1) midPoint +1; 2) midPoint - 1;
- [ ] None of the above

54

```
push(5)
push(8)
pop
push(2)
push(5)
pop
pop
pop
push(1)
pop
```

Choose the correct output for the following sequence of operations? Assume that pop print the top of the stack.
(2 Points)

- [ ] 8 5 2 5 1
- [ ] 8 5 5 2 1
- [ ] 5 8 2 5 1
- [x] None of the above

55

The algorithm for deleting from an unsorted list has the last item replace the item being deleted.
(2 Points)

- True
- False

56

Given only the external pointer to a linked list, it is faster to insert a node at the front of the list than at the back.
(2 Points)

- True
- False

57

Two-dimensional arrays are stored in column order in C++.
(2 Points)

- True
- False