4장 단순연결리스트 연습문제

1 ➊과 ➋에 알맞은 용어를 각각 채우시오.

|  |
| --- |
| ➊는(은) 배열에 데이터를 차례대로 저장하므로 데이터의 실제 위치 순서로 데이터가 구성된다.  ➋에서는 데이터를 노드 단위로 삽입/삭제한다. |

2 선형 리스트와 비교한 단순 연결 리스트에 대한 설명이다. 거리가 먼 것은?

① 물리적으로 연결되어 있지 않은 데이터가 연결된다.

② 연결을 위한 링크(Link)가 필요하다.

③ 중간에 새로운 데이터를 삽입할 때는 비효율적이다.

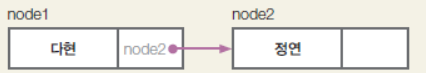
④ 노드(Node)를 사용해서 데이터를 표현한다.

3 노드 구조에서 ⓐ와 ⓑ를 무엇이라고 하는지 다음 중에서 고르시오.

|  |
| --- |
| 리스트, 링크, 헤드, 배열, 주소, 번지, 데이터 |



4 그림과 같이 노드를 생성하고 연결하는 코드를 차례대로 올바르게 나열한 것은?



|  |
| --- |
| ⓐ node1.data = "다현" ⓑ node1.link = node2  ⓒ node1 = Node() ⓓ node2.data = "정연"  ⓔ node2 = Node() |

① ⓔ-ⓓ-ⓑ-ⓐ-ⓒ

② ⓒ-ⓐ-ⓔ-ⓓ-ⓑ

③ ⓔ-ⓐ-ⓒ-ⓑ-ⓓ

④ ⓐ-ⓑ-ⓒ-ⓓ-ⓔ

5 단순 연결 리스트의 맨 앞 데이터를 삭제하는 코드를 차례대로 나열하시오.

|  |
| --- |
| ⓐ head = head.link  ⓑ current = head  ⓒ del(current) |

6 다음과 같이 코드가 작성되어 있을 때, 쯔위를 출력하는 코드로 올바른 것은?

|  |
| --- |
| node1 = Node(); node1.data = "다현"  node2 = Node(); node2.data = "정연"; node1.link = node2  node3 = Node(); node3.data = "쯔위"; node2.link = node3 |

① print(node1)

② print(node1.data)

③ print(node1.link.data)

④ print(node1.link.link.data)

7 단순 연결 리스트의 맨 앞에 야옹이 데이터를 삽입하는 ➊~➌에 들어갈 코드를 다음 중에서 고르시오.

|  |
| --- |
| nodeNode(), head, "야옹이", data, node.data, link, node.link |

|  |
| --- |
| node = Node()  ➊ = "야옹이"  ➋ = head  head = ➌ |

8 노드에서 데이터를 검색하는 함수다. ➊에 들어갈 코드를 고르시오.

|  |
| --- |
| def findNode(findData) :  global memory, head, current, pre  current = head  if current.data == findData :  return current  ➊ :  current = current.link  if current.data == findData :  return current  return Node() |

① while current.link == None

② while current.link != None

③ while head.link != None

④ while head.link == None