

Risposte Esame di Programmazione - 2021-12-22

Esercizio 1A

Realizzare la struct CIRCLE (centro e raggio) e la struct POINT necessaria per realizzare il centro.

```
struct Point {  
    double x, y;  
};  
  
struct Circle {  
    Point centro;  
    double raggio;  
};
```

Esercizio 1B

Realizzare una funzione che verifichi se due cerchi hanno lo stesso perimetro.

```
bool stessoPerimetro(Circle c1, Circle c2) {  
    return (2 * M_PI * c1.raggio) == (2 * M_PI * c2.raggio);  
}
```

Esercizio 1C

Realizzare una funzione che verifichi se, dati due cerchi, uno è contenuto nell'altro.

```
bool contenuto(Circle c1, Circle c2) {  
    double distanzaCentri = sqrt(pow(c1.centro.x - c2.centro.x, 2) + pow(c1.centro.y - c2.centro.y, 2));
```

Risposte Esame di Programmazione - 2021-12-22

```
return distanzaCentri + c1.raggio <= c2.raggio || distanzaCentri + c2.raggio <= c1.raggio;  
}
```

Esercizio 2A

Realizzare una funzione che preso un $n > 0$ restituisca un vector contenente (esattamente) i primi n multipli di n .

```
std::vector<int> primiMultipli(int n) {  
    std::vector<int> multipli;  
    for (int i = 1; i <= n; ++i) {  
        multipli.push_back(i * n);  
    }  
    return multipli;  
}
```

Esercizio 3A

Realizzare una funzione ricorsiva che restituisca la lunghezza della lista.

```
int lunghezzaLista(lista l) {  
    if (l == NULL) {  
        return 0;  
    } else {  
        return 1 + lunghezzaLista(l->next);  
    }  
}
```

```
}
```

Esercizio 3B

Realizzare una funzione che restituisca true se tutti gli elementi della lista sono "ecco", false altrimenti. Trattare in modo opportuno il caso lista vuota.

```
bool tuttiEcco(lista l) {  
    if (l == NULL) {  
        return true;  
    } else {  
        return (l->str == "ecco") && tuttiEcco(l->next);  
    }  
}
```