

Modalità di consegna dell'elaborato

Consegna tramite email con mittente, destinatari e oggetto come segue:

- **MITTENTE:** vostro indirizzo di posta elettronica del dominio studenti.unitn.it o unitn.it
- **DESTINATARIO:** lorenzom<dot>gramola<at>gmail<dot>com
- **CC:** paola<dot>quaglia<at>unitn<dot>it
- **OGGETTO:** LFC-LAB-PRJ **id**, dove **id** è il vostro numero di matricola
es: se la vostra matricola è 123456 allora l'oggetto è LFC-LAB-PRJ 123456

Inoltre

- dovrete allegare alla mail il progetto zippato con nome "**prjid**"(stesso esempio di cui sopra: prj123456). Il formato del progetto zippato deve esser **.zip** (non tar, rar, 7zip etc etc)
- una volta scompattato il file compresso dovrà creare la sua cartella, il cui nome sarà il vostro numero di matricola.
- il file zippato non dovrà contenere nessun eseguibile e nessun file oggetto (nessuno, compresi gli eventuali permessi di esecuzione assegnati ai file)
- **ogni file** dovrà contenere come prima riga il vostro numero di matricola
- prendete come base di partenza l'esercizio 5.2; il file zippato dovrà contenere solamente i file necessari alla build (lexer, parser, eventuali header e file di implementazione, in nessun caso è ammesso il logger che è a vostro unico utilizzo)
- oltre ai file necessari alla build dovrà essere fornito un Makefile con le operazioni di build del progetto e di clean dei file creati (il file eseguibile creato dovrà essere chiamato "interpreter"), con opzione -std=c99 per la compilazione.

Il rispetto dei requisiti suddetti è considerato condizione necessaria alla validazione del progetto.

Controllate puntualmente ciascun requisito, non sono ammesse consegne multiple per lo stesso appello (il che implica l'invio di una sola email).

Aggiungere al linguaggio visto durante le lezioni di laboratorio il passaggio di parametri per riferimento usando `byRef` come parola chiave per identificare tale modalità.

Il passaggio di parametri per riferimento è un tipo di passaggio di parametri che invece di eseguire una copia dei valori degli argomenti passati alla funzione utilizza un puntatore ai valori stessi dei parametri. Questo implica che il valore di un argomento cambiato all'interno della funzione si rifletterà al di fuori del suo scope, modificando quindi il valore del parametro passato (solo per gli argomenti della funzione formalmente dichiarati `byRef`).

Un esempio di input, per aiutarvi a svolgere l'esercizio è il seguente:

```
copyInto(int x, byRef int y){
    print x;
    print y;
    y = x;
    print x;
    print y;
}

main
{
    int a;
    int b;
    a = 7;
    b = 13;
    print a;
    print b;
    eval copyInto(a,b);
    print a;
    print b;
}
```

Tale esempio stamperà 7 e poi 13.

Chiamerà la procedura che stamperà 7 e 13, e poi 7 e 7.

All'uscita della funzione il main finirà l'esecuzione stampando 7 e 7.

Prendete di riferimento la cartella input per controllare che tutti i precedenti file di input mantengano la loro coerenza rispetto alle modifiche richieste.

NOTA BENE:

L'esempio di guida proposto in precedenza è a titolo esemplificativo. Ottenere il risultato desiderato non implica l'esito positivo della prova di sbarramento, ma solo il fatto che il lavoro si sta svolgendo nella giusta direzione.

Buon lavoro.