

## Indicazioni generali sul progetto di Architetture II

### Cosa dovrebbe esserci:

- Il progetto deve implementare una procedura ricorsiva.
- Il progetto deve gestire un array di dimensione potenzialmente infinita possibilmente non di elementi di dimensione di una word.
- Il progetto deve essere ben commentato. I commenti dovrebbero spiegare cosa fa un blocco funzionale. Evitate la parafrasi del testo: chi leggerà il vostro tabulato sa perfettamente cosa significa e come si invoca una syscall 4 o cosa significa “addi \$t0,\$t0,1”.
- Il progetto deve essere corredato da una **relazione** esauriente quanto compatta, **rigorosamente in formato PDF**. Mettete:
  - le specifiche, cioè cosa fa il programma, che input accetta, cosa deve sapere l'utente per poterlo usare.
  - uno schema generale (logico, non il tabulato del *main*) con i blocchi funzionali del program
  - una spiegazione a grandi linee che descriva come il circuito funziona e come i blocchi interagiscono tra loro
  - la spiegazione dettagliata delle parti salienti del programma.
  - le conclusioni in cui in una decina di righe spiegate cosa avete fatto e come volevate andare avanti nello sviluppo se ne aveste avuto il tempo e la possibilità hardware: esempio “avendo a disposizione un lettore di schede perforate si poteva aggiungere una periferica di input ausiliaria per semplificare l’inserimento.”
- La relazione può essere strutturata seguendo questa traccia (**siate concisi!**):
  - Introduzione generale con specifiche e descrizione delle tecnologie utilizzate per realizzare il progetto, esempio: “ho implementato una calcolatrice che usa la notazione polacca inversa”.
  - schema generale logico e spiegazione delle interazioni tra i vari blocchi
  - dettaglio delle parti importanti (non le parti triviali o di input/output a meno che non siano importanti al fine dell'algoritmo).
  - Conclusioni e possibili sviluppi futuri

### Cosa non dovrebbe esserci:

- Evitate di realizzare i seguenti progetti in quanto ampiamente trattati dai vostri colleghi:
  - calcolatrici
  - riconoscitori di stringhe palindrome.
  - Algoritmi che verificano le parentesi
  - algoritmi che calcolano solo fibonacci o il fattoriale.
- Evitate di realizzare progetti presenti nel sito del corso o *ispirati troppo strettamente* a progetti di vostri colleghi.
- Evitate di realizzare progetti troppo semplici o mastodontici, ai limiti della potenza di elaborazione del PC o della comprensione umana. Il progetto valuta soprattutto le vostre capacità di organizzazione e sviluppo più che la vostra originalità/abilità di programmatori/sviluppatori.
- Evitate di realizzare documentazioni enciclopediche contenenti spiegazioni banali o inutili. Siate essenziali.

## Cosa succederà alla verifica del progetto

La correzione sarà molto simile a questa sequenza:

- se ci sono problemi con l'interaccia (tipo non fa quello che deve non si capisce come usarla, etc..) si guarda insieme.
  - se si scopre che era il docente che non aveva capito si prosegue.
  - altrimenti ci si vede la prossima volta.
- poi domanda "*mi parli brevemente del programma*" (evitate di descrivere aspetti non attinenti al circuito tipo "come avete avuto l'idea", "la strategia migliore per vincere in questa simulazione di gioco e'..." , "mentre lo realizzavo ho pensato che sarebbe molto bello realizzare un altro circuito che fa il contrario" )
- poi domanda "*cosa serve questo, come ha fatto a farlo*" (dito puntato in una parte del tabulato)
  - spesso capita che una vostra soluzione non sia ottimale o proprio non viene usata nella pratica usuale. Questo vi verrà fatto notare sempre ed avrà un effetto sul vostro voto solo se la soluzione da voi adottata non è ragionevole, esempio una parte di circuito che genera un segnale duplicata per ottenere lo stesso segnale negato.
- poi domanda "*come si poteva fare in maniera diversa, usando ad esempio ..., come veniva? mi faccia un breve schizzo di come intende progettarlo*"
- infine voto "*le propongo xy, le va bene?*"
  - Si -> non ci si vede più.
  - No-> viene la prossima volta.

## Cosa non va detto alla verifica

- "Prof": meglio "professore" o "docente" o meglio niente, date del Lei e basta.
- "Ma sul mio PC funzionava": i progetti devono avere un minimo di portabilità e stabilità.
- Manca la relazione in un qualunque formato, cartaceo o elettronico e lo studente risponde:
  - "Il cane mi ha mangiato la relazione ma l'ho fatta": indice di disordine nella migliore delle ipotesi.
  - "Ho lasciata a casa la relazione ma l'ho fatta": come sopra.
- "Pensavo che fatto così fosse giusto": indice di scarsa collaborazione con i suoi colleghi e con il corpo docente.
- "All'ITIS l'abbiamo fatto diverso": questa è l'Università, altro ambiente, altre regole.
- "Non mi ricordo a cosa serve" sinonimo di "l'ho copiato/comprato ma non l'ho studiato abbastanza"
- "L'ho fatto tre mesi fa, non mi ricordo benissimo come funziona" sinonimo di "son venuto qui senza studiare/prepararmi perchè quest'esame non mi interessa, anche se non lo passo è uguale"

## Modalità di consegna e valutazione della parte di laboratorio

1. I progetti vanno concordati anticipatamente inviando un breve riassunto di quanto si intende realizzare all'indirizzo [marchi@di.unimi.it](mailto:marchi@di.unimi.it).
2. I progetti devono seguire le indicazioni generali indicate in questo documento.
3. I progetti vanno consegnati entro fine settembre 2014. La relazione in formato *pdf* ed il progetto in formato *asm* devono essere impaccati in un unico ZIP/RAR/7z file e spediti all'indirizzo [marchi@di.unimi.it](mailto:marchi@di.unimi.it) indicando nell'oggetto "Consegna progetto architetture II 2013/2014 matr. <il vostro numero di matricola>". **Ogni file deve riportare il vostro numero di matricola.**
4. La discussione del progetto avverrà in data da concordare via mail.
5. La valutazione verrà fatta rispetto a questi parametri (elencati in ordine decrescente di importanza):
  - Coerenza e pulizia dell'algoritmo adottato: quanto le soluzioni adottate siano adatte per implementare le varie funzioni del progetto e la chiarezza del layout realizzato.
  - Correttezza nella realizzazione della specifica dichiarata: se il programma realizzato fa quello che doveva fare.
  - Efficacia espressiva della relazione: relazioni compatte ma esaustive dei punti salienti sono da preferire a relazioni esaustive di ogni dettaglio ma enciclopediche.
  - Originalità del progetto.
  - Discussione orale