

LIVE SA LABOSA:

Uvodno ispitivanje: podijeljeni ste u 2 grupe.

A- treba prvi program prepraviti tako da ne radi sa fiksom vel. bloka. Treba stati s prebacivanjem kad procita 100

B-kako ste morali brojati neparne brojeve. Sad morate parne(sad nezz jel i parne i neparne ili samo parne)

---

live s labosa: nama su eliminacijska pokazi ispis memorije i jedno pitanje tipa kako staviti brkpt

---

eliminacijska s labosa u 14 (oboje u 1.zadatku)

A grupa: učitava dok ne naiđe na parni broj

B grupa: izbrojati među početnim brojevima koliko ih je parnog pariteta

---

Eliminacijska s labosa: pokaži registar koji sadrži adresu memorijske lokacije s koje učitavaš podatke.

---

Jutros na labosu:

Od svih je tražio simulacije obadva zadataka u isto vrijeme i onda je išao pojedinačno i ispitivao eliminacijska. Mislim da je samo jednog poslao van, ali mu nije dao -2 boda nego 0 bodova, ali mu nije dao da piše blic.

Moja eliminacijska:

Pitao me 2. zadatak (kod na slici):

1. Što je uopće bio zadatak? Objasnite.
2. Objasnite kako vam radi potprogram PAR.
3. U kodu, za šta vam služe `AND R0, %B 01111111, R0 (linija 47)` i `OR R0, %B 10000000, R0 (linija 50)`.
4. Zašto baš `SP+10` u liniji 29?
5. Nacrtajte stog nakon `LOADB R0, (SP+10) (linija 29)`

---

live s labosa eliminacijska izmjeniti kod:

grupa a: prvi zad dodati uvjet da je broj paran, znaci mora mu biti negativan predznak i paran da bi se komplementirao

grupa b: drugi zadatak prebrojati parne brojeve

---

---

Lab u 14.00 -Asistenti su trazili da se pokrenu simulacije. Onda su isli redom i ispitivali svakoga ponesto.

Mene je pitao potprogram, kako prenosim rez u poptpr u program i sa sp+10. Zasto bas 7 kod rotacija. Koliko bi bilo rotacija da je polurijec u pitanju (15). Pokazite mi ispis programa na xcompasu u memoriji. Zasto je ispis 000000xx, misli se na simulaciji one mem lokacije kad ispisuje vrijednosti. I eto to je koliko se sjecam

---

Pitanja koja sam dobio:

2 zadatak

Pokažite gdje je u memoriji zapisan BLOK?

Podatak F3 u što se on pretvara? I pokaži gdje je u memoriji

Pokaži liniju s kojom spremaš taj podatak u mem?

Pokažite od kuda do kuda je potprogram?

Šta radi naredba RET?

Kako zna gdje se vraća?

Pokaži kako izgleda stog prije LOADB R0, (SP+10) i zašto baš +10

---

Live update sa labosa sada u 10 sati. Krcata učionica, 2 asistenta koja dolaze do vas kako ih tko zove da je spreman prezentirati te zadatke. Casual atmosfera, puno buke. 🇺🇸

Ja se prvi javio, pitao me u prvom zadatku da pokažem memorijske lokacije od 0002 (drugi rezultat koji treba dobiti na lokaciji 1000). Pokazao mu, rekao da je little endian i to je bilo to. Kaže "Prvi ste pa neće biti teško." Asistent je Daniel Hofman.

---

Neka od mojih pitanja ...

Kako bi ispitali stanje PC registra?  
(break point, display all registers)

Čemu služi/što je SR?

Čemu služi ovaj dio programa?

XOR R3, -1, R3

ADD R3, 1, R3

---

---

---

Asistenti su bili više nego fer i korektni. Jedan momak nije znao nešto što ga je pitao pa ga je asistent pustio da se malo smisli dok on ispita druge.

Dobijete 2 zadatka gdje je opisano šta trebaju raditi i kod sa par crtica koje trebate popuniti. Sve je dobro komentirano.

---

---

Prvi zadatak je bio prebacivanje 16-bitnog 2'k broja u 8 bitni 1'k s tim da prvi član niza predstavlja broj članova u tom nizu.

Trebalo je popuniti naredbu gdje učitavate taj brojač iz memorijske lokacije, u slučaju kad je broj negativan trebalo je napisati naredbu da samo od ovog 2'k oduzmete 1 da dobijete broj u 1'k, povećati pokazivač na memorijsku lokaciju za spremanje sljedećeg 8-bitnog broja, s tim da je trebalo uvećati samo za 1 jer spremate 8 bitne brojeve. Jedna naredba je bila pogrešno napisana, pisao je logički pomak za 8 a trebalo je za 16.

Drugi zadatak je bio sa stogom, trebalo je provjeriti koliko je bitova na istom mjestu u 2 broja jednako. Trebalo je napisati naredbu za čišćenje stoga poslije poziva potprograma, naredbu za rotaciju maske, spremanje rezultata na fiksnu memorijsku lokaciju REZ i inicijaliziranje brojača u potprogramu gdje ste trebali prepoznati koji od registara u zadatku obavlja tu ulogu.

---

---

Evo zadnja grupa danas, od 18-20 u A101. Labos prošao u revijalnom tonu. Mene pitalo breakpoint i pokazat memoriju u prvom zadatku. U drugom malo objasniti potprogram što radi.

Skoro su svi došli do blica, da bi uspio biti izbačen trebao si zaista pokazati zavidnu razinu neznanja. Najteže pitanje koje sam primjetio jest da se objasne malo pushevi i stog i gdje se nalazi SP u kojem trenutku.

A i imate 15min samo, nije komplicirano za onoga tko je ajmo rec' u tom štosu i koji je dosta porgiramirao u assembleru.

a sam trebao imati 3,5 ali nisam stavio %D kod brojaca petlje, jer nisam vidio da je 16b podatak, i jos poneka linija jer je dosta tesko skuziti sta oni tu rade u tako malo vremena.

Ovo vam je najtezi blic, ostali su lakse (vanjske i prekidi)

live report sa labosa, imam nula bodova sigurno 🍷

evo pitanja koja sam cuo da je asistent postavio: Pokazati rez blok u prvom zadatku. objasniti u drugom zast je sp+10. sta radi naredba loadb.

Pokazati rez.blok u drugom zadatku. Pokazati od kud do kud ide petlja u prvom zadatku. zasto u drugom imamo na pocetku potprograma ona 3 push-a i zasto je na kraju kod ona 3 pop-a obrnut redoslijed registara.

---

---

EDIT: test za bodove bio dosta slican samim zadacima,u prvom zadatku su se trebale 32 bitne rijeci u 1'k formatu sa lokacije 300 spremati u 16 bitne 2'k rijeci na lokaciji 400). trebalo je dopisati 3-4 linije koda i ispraviti gresku (lokacije gdje spremamo polurijeci pomicemo sa add za 2,a ne za 4 kako su oni napisali) . drugi zadatak je mozda cak i isti kao drugi odavde:

<http://www.fer2.net/showpost.php?p=1...&postcount=229> ) i trebalo je 4 linije koda dopuniti. lijepo vam oni sa strane iskomentiraju skoro svaki red pa se da lakse pratiti sta i kako program radi. sretno :)

Ovaj put su bila dva zadatka slicni onima za labos (ne ono za pripremu nego ona dva zadatka za labos), s malim preinakama (tipa nešto s riječima od 16 bita i nešto da komada podatke i ispituje neki bit).

Dobiješ opis zadatka, kod s komentarima i par mjesta za ubaciti kod kako bi program radio to što se gore traži.

Prvi zadatak je bio kraći i trebalo je dodati 2 linije i naći jednu grešku, a drugi je bio malo dulji i mislim da je bilo za dodati 4 linije.

S tim da kod nekih linija imaš kometar koji ti pomogne.

Piše se na papiru.

Reporting live

Prvi zadatak:

1. zasto u brojac stavljamo 504 a ne 500?
2. citanje podataka na odredisnom bloku tj provjerio je jel znam da je little endian

Drugi zadatak:

1. Kojom petljom i na koji nacin odredujemo paritet podatka

1. Zadatak- bio je napisan kratak kod (pretvara 1'k u 2'k) i treba pronaći jednu grešku (greška je bila u uvjetu- umjesto jr\_NC trebalo je biti jr\_NN) i u tom zadatku je bilo 3 ili 4 prazne linije i trebalo je upisati naredbe da program bude ispravan.

2. Zadatak- zadatak sa potprogramom, trebalo je samo popuniti prazne linije da kod bude ispravan.

Nemojte se uplašiti ako je kod preko cijele stranice. Bio je zadatak nešto sa niblovima (to su grupe od po 4 bitova koliko sam skužio)

Zapamtio sam iz prvog zadatka da morate promijeniti iz JR\_Z u JR\_NN, kod negativnog morate upisati XOR R3(ja mislim da je R3),-1,R3

Na 2. zadatku imate prvu liniju MOVE 10000, SP... Evo 1.5 bodova od mene :)

edit:

1. zadatak je bio 32-bitni 1'k u 16-bitni 2'k , greska je u JR\_C,treba bit JR\_P ili JR\_NN,ovo ostalo zbilja nije tesko,tipa smanjivanje brojaca i takve stvari

2.potprogram prima 2 parametra sa stoga i racuna broj bitova koji su isti u oba podatka (znaci na istoj poziciji u oba podatka se nalazi ista vrijednost) ,te taj broj vraca preko pozicije u memoriji.Ovo je bilo malo tricky.Znam da je prva linija bila MOVE 400(il tak nest) R2(il tak nest 🍷),druga PUSH R0.Zatim u potprogramu radi sljedece XOR R1,R2,R3 (r1 i r2 su argumenti),zatim AND R3, 1, R3, i onda linija gdje treba pisat JR\_NZ RAZLICITI, il nest toga tipa.Uglavnom, fora je u tome da uspoređuje bit po bit.Kad ih XORa, na zadnjem ce mjestu bit 1 ako su razliciti,a 0 ako su isti. A kad to sve ANDa s 00...001,ako dobije 1 onda znaci da su bitovi razliciti,dakle treba skocit na labelu RAZLICITI gdje se samo povecava brojac petlje, a preskace dio gdje se povecava i brojac istih bitova.I na kraju je jos nesto tipa STORE R3, (REZ).

1. zadatak (2. nesto)- dobis 32 bitni 1'k broj i moras spremit kao 16 bitni broj s predznakom. Radi jednostavnosti broj nikad nece premasiti opseg 16 bitnih brojeva s predznakom. Trebas:

-smislit kako od 32 bitnog 1'k broja dobit predznak s time da je pocetna nareba pogresno napisana (pise OR R2, R2, R2, moras ju prekrizit i napisat svoju). Ja sam npr koristio AND R2, FFFF0000, R3 i gledao jel rez 0 ili 1. Moze se na razne nacine inace rijesit.

-povecat pokazivac na ishodisne podatke koji su 32 bitni (ADD R2, 4, R2)

- jos 2 linije koje se nemrem sjetit

2. Zadatak (13. C): dobis 32 bitni broj, moras ga preuredit tako da ako je prvi bit pojedinog nibbla (4 bita)=0 prebrises taj nibble s 0, inace ga ostavis. To preuredivanje ide u potprogramu, a on funkcionira tako da dve maske rotira malo po malo po cijelom broju u petlji. Primjer F5758611 -> F5008600. (valjda je tak nekak iso). Trebas:

-inicijalizirat stog (MOVE 10000, SP)

-dovrsit JR petlje (JR\_uvjet LOOP, uvjet treba skuzit u zadatku)

-jos 2 linije koje se nemrem sjetit

---