**BRAC UNIVERSITY**

A

**Department of Computer Science and Engineering**

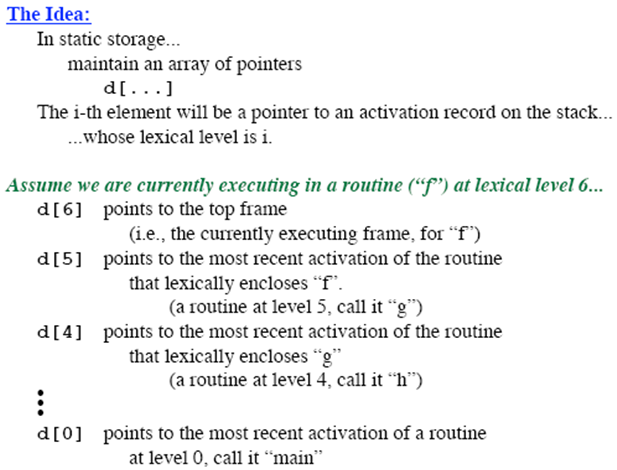
**CSE420: Compiler Design**

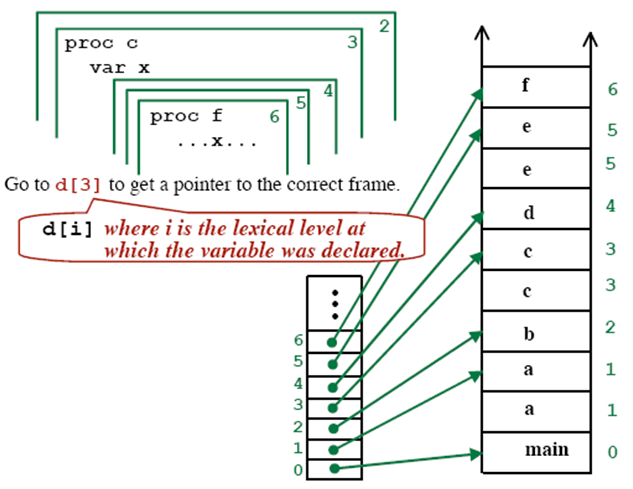
**Quiz 03, Spring 2016**

**Duration: 1.00 hours, Total Marks: 30**

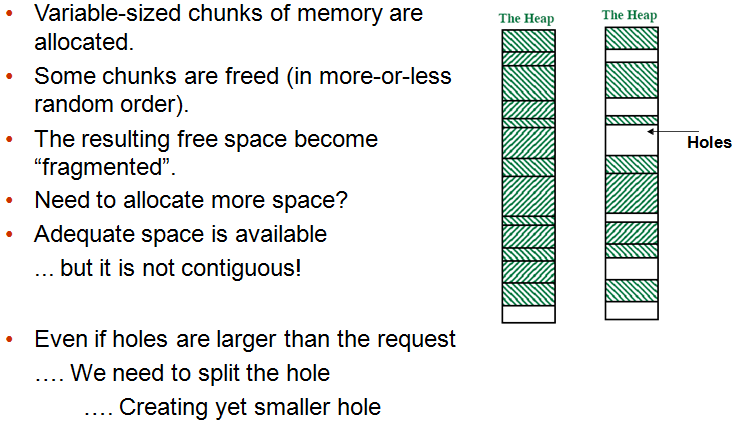
|  |
| --- |
| Student Name:  Student ID:  Section: |

1. Draw a block diagram showing the caller-callee responsibilities in the construction of activation record. [5]
2. Define display register with appropriate example. [3]





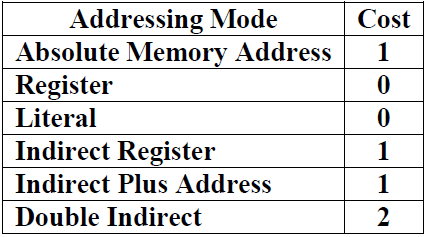
1. What are the basic reasons behind fragmentation? [2]



1. Name the main design issues of code generator. [1]

* Instruction Selection
* Register Allocation
* Evaluation Order

1. Consider a hypothetical machine with four registers R1, R2 and six addressing modes with the following costs.



Calculate the total cost of following code segment. [4]

*a=b+c*

*d=g+t*

*param a*

*param d*

*call s, 2*

**move b,R1**

**add c,R1**

**move g, r2**

**add t, r2**

**param r1**

**param r2**

**call s, 2**

1. Cost =1+1+0+0=2
2. Cost=1+1+0+0=2
3. Cost=1+1+0+0=2
4. Cost=1+1+0+0=2
5. Cost=1+0+0+0=1
6. Cost=1+0+0+0=1
7. Cost=1+1+0+0=2

Total Cost = 12

1. Determine the set variable used and defined by following code fragment and also define the live variable at point of the code. [4]

j = 4

k = j + 1

j = 6 *(Point 01)*

m = k \* j *(Point 02)*

m = m + 2

k = j + l*(Point 03)*

j = k + j

Use ={l}

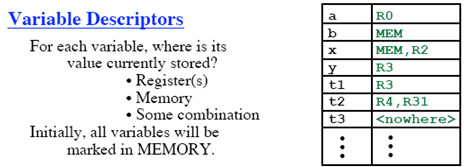
Def={j,k,m,k}

Point 01, Live = {k}

Point 02, Live = {j,k}

Point 03, Live = {j,l}

1. Define variable descriptor with example. [2]

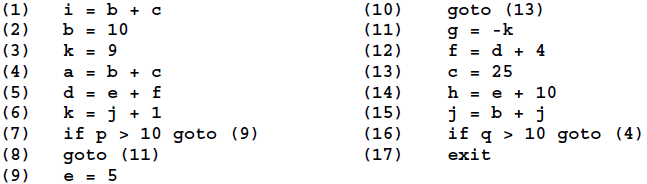


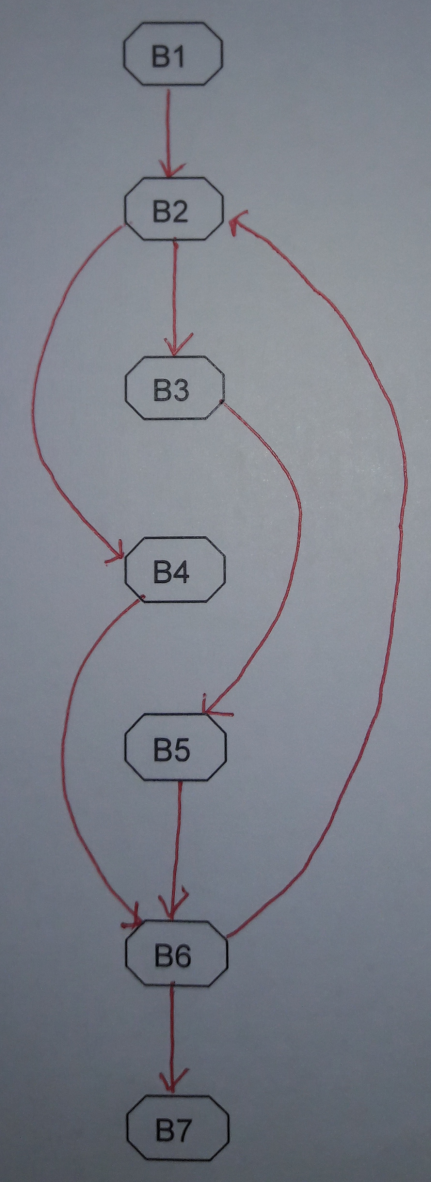
1. Define Peephole Optimization. [2]

Pass over generated code to examine a few instructions, typically 2 to 4

* + Redundant instruction Elimination: Use algebraic identities
  + Flow of control optimization: removal of redundant jumps
  + Use of machine idioms

1. Determine the control flow graph of following instruction set. [4]





1. Following is the intermediate code to compute the dot product of two vectors A and B. Optimize this code by eliminating unnecessary operations and variables. [3]

