# General Computer Science II (320201) Spring 2009

Michael Kohlhase
Jacobs University Bremen
FOR COURSE PURPOSES ONLY

April 8, 2013

## Contents

# Quiz 1: Graphs and Trees
## (Given Feb. 9.)

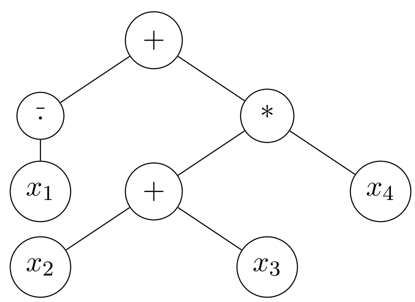
12pt

**Problem 1.1 (Parse trees and isomorphism)**
Let $P_e$ be the parse-tree of $e := \overline{x_1} + (x_2 + x_3) * x_4$

1. Draw the graphic representation of $P_e$.
2. Write the mathematical representation of a graph $G$ that is different but equivalent to $P_e$.

---

**Solution:**

1. 


2. $G := \langle\{A, B, C, D, 1, 2, 3, 4\}, \{\langle A, B\rangle, \langle A, C\rangle, \langle B, 1\rangle, \langle C, D\rangle, \langle C, 4\rangle, \langle D, 2\rangle, \langle D, 3\rangle\}\rangle$

---

# Quiz 2: Positional Number Systems
## (Given Feb. 16.)

12pt

**Problem 2.2 (Playing with bases)**
Convert 2748 from decimal to hexadecimal, binary and octal representation.

---

**Solution:** Divide repeatedly 2748 by 16 and take the remainders - get $ABC$. $A = 1010$, $B = 1011$ and $C = 1100$, thus $ABC = 101010111100$. Starting from right to left convert every 3 bits to their corresponding octal value: 5274.

---

# Quiz 3: Adders and TCN
## (Given Feb. 23.)

6pt

**Problem 3.3 (Adding TCN numbers)**
The numbers 16 and 7 are added using an adder:

1. What is the minimum number of bits necessary to represent these numbers in two's complement notation?
2. What is the minimum depth of a Carry Chain Adder that can sum these numbers?

---

**Solution:**

1. 6
2. $6 \cdot 3 = 18$

---

**Problem 3.4 (Two's complement conversion)** 6pt

1. Convert $-42$ to 12-bit two's complement.
2. Convert the two's complement number 111110110101 to decimal.

---

**Solution:**

1. $42 = 0101010$, flip all bits $\rightarrow 1010101$, add $1 \rightarrow 1010110$, duplicate sign bit until 12 bits $\rightarrow 111111010110$
2. 111110110101, subtract one $\rightarrow 111110110100$, flip all bits $\rightarrow 000001001011 = 75$.

---

# Quiz 4: Memory
## (Given Mar 2.)

12pt

**Problem 4.5 (3-bit Address Decoder)**
Design a three-bit address decoder. Draw your circuit as a graph.

# Quiz 5: Virtual Machine

## (Given March 9.)

12pt

**Problem 5.6 (a mod b)**
Assume the data stack is initialized with `con` $a$ and `con` $b$ for some natural number $a$ and $b$. Write a $\mathcal{L}$(`VM`) program that returns on top of the stack the value of $a \, mod \, b$.

---

**Solution:**

```
con a
con b
peek 0
peek 1
leq
cjp 8 ; if b¿a
peek 1 ; then a=a-b
peek 0
sub
poke 0
jp −14 ; jump to third line
peek 0 ; else push a
halt
```

---

# Quiz 6: SW Language
## (Given March 24.)

12pt

**Problem 6.7 (SW program)**
Write a program in Simple While Language which is equivalent to the following program in abstract syntax:

```
var i := 0; var a := 1; var b := 1; var f := 0;
while (i <= 10) do begin
   f := a + b;
   a := b;
   b := f;
   i = i + 1;
end
return (a + b) ∗ f;
```

# Quiz 7: Static Procedures
## (Given March 16)

12pt

**Problem 7.8 (Towers of Hanoi)**
The Towers of Hanoi is a very famous mathematical puzzle. Given three pegs, one with a set of $N$ disks of increasing size, determine the minimum (optimal) number of steps it takes to move all the disks from their initial position to another peg without placing a larger disk on top of a smaller one.

Write down a static procedure in $\mathcal{L}$(VMP) that computes the solution to the puzzle, given $N$.

---

**Solution:**

```
proc 1 27

;;if N=1 jump to the end
con 1 arg 1 sub cjp 16

;;h(n-1)*2+1
con 1 arg 1 sub
call 0 con 2 mul con 1 add return

;;return 1 for N=1
con 1 return
```

---

# Quiz 8: TM and Problem formulation

## (Given March 30)

**Problem 8.9 (OR the Tape)** 6pt

Design a TM that implements the n-ary OR operator on its tape: Started with a sequence of 0s and 1s on the tape, it writes the results at the end of this input and halts. For example, a tape with 111 on it will be transformed in 1111. Your TM needs to have at most 3 states, halting state included. Also, what you TM returns on the empty input is not important.

---

**Solution:**

```
1,_ H,0,>
1,1 2,1,>
1,0 1,0,>
2,_ H,1,>
2,1 2,1,>
2,0 2,0,>
```

---

# Quiz 9: Informed Search
## (Given April 20)

6pt

**Problem 9.10** ($A^*$ **Theory)**
What is the condition on the heuristic function that makes $A^*$ optimal? Does a heuristic with this condition always exist?

---

**Solution:** Admissible heuristic - always underestimates the real cost to the goal. This always exists: $h(x) = 0$.

---

# Quiz 10: Local Search
## (Given April 27)

6pt

**Problem 10.11 (Local Beam Search)**
What known algorithm does Local Beam Search become if $k = 1$?

---

**Solution:** The beam is 1, so there is only one starting point, so Hill Climbing.

---

**Problem 10.12 (Greedy vs. Hill Climbing)** 6pt

What is the fundamental difference between Greedy Search and Hill Climbing? Explain.

---

**Solution:** HC is local search, i.e. the path is not kept because we are only interested in finding a solution and not how we got there.

---

# Quiz 11: Prolog
## (Given May 4)

4pt

**Problem 11.13 (Knowledge Base)**
What is the definition of a knowledge base?

---

**Solution:** The knowledge base given by a set of facts and rules is that set of facts that can be derived from it by Modus Ponens ($\Rightarrow I$) and $\wedge I$.

---

**Problem 11.14 (Reachability in ProLog)** 8pt

In a graph, it is sometimes important to be able to tell if a given node is reachable from another node. Given a predicate $edge(X,Y)$ that is true if there is an edge between nodes $X$ and $Y$, define a predicate $reachable(X,Y)$ that is true if there is a path from $X$ to $Y$, or if $X = Y$.

---

**Solution:**

```
reachable(X,X).
reachable(X,Y):−edge(X,Y).
reachable(X,Y):−edge(X,Z), reachable(Z,Y).
```

---

# Quiz 12: Prolog
## (Given May 11)

**Problem 12.15 (Arithmetics in ProLog)** 8pt

- Give the definition of a ProLog rule triangle(A,B,C) that is true only if $A$, $B$ and $C$ can be the three sides of a triangle. You can assume that the three numbers are positive.
- What will ProLog return if you write the query ?−triangle(1,2,C).. Explain briefly.

---

**Solution:**

```
triangle(A,B,C) :- AB is A+B, BC is B+C, AC is A+C, A < BC, B < AC, C <AB.
```

The query ?−triangle(1,2,C). will result in an error since in BC = B+C, C will not be instantiated.

---

**Problem 12.16 (Unification)** 4pt

Give a most general unifier of the terms $\mathbf{A} = f(X, g(Y, X))$ and $\mathbf{B} = f(Y, Z)$. Give one more unifier that is NOT most general.

---

**Solution:**

- $[Y/X], [g(Y, Y)/Z]$
- $[a/X], [a/Y], [g(a, a)/Z]$

---