Principles of Programming Languages
Answers for small examination 2

Problem 1 Show the type consistency of the following program fragment, which is written in the subset of C language presented in the lecture, according to (1) and (2).
(Answer)

```
int *p;
int x[3];
p = x;
```

(1) Rewrite the variable declarations `int *p;` and `int x[3];` in the postfix notation presented in the lecture.

(Answer)

```
p : int *
x : int [3]
```

(2) Show the type consistency of the assignment expression `p=x` by applying the inference rules to the declarations of `p` and `x` in the postfix notation obtained in (1).

(Answer)

$$\dfrac{\texttt{p : int} * \qquad \dfrac{\texttt{x : int [3]}}{\texttt{x : int \&}}}{\texttt{p} = \texttt{x : int \&}}$$

Problem 2 A lambda expression $(\lambda x.\ \lambda y.\ x)\ ((\lambda z.\ z)\ w)$ can be transformed to $(\lambda y.\ w)$ by applying the β reductions. Write the each step of the β reductions. (Although there are more than one sequences of β reductions, write one of them.)
(Answer 1)

$$(\lambda x.\ \lambda y.\ x)\ ((\lambda z.\ z)\ w) \xrightarrow[\beta]{} (\lambda x.\ \lambda y.\ x)\ w \xrightarrow[\beta]{} \lambda y.\ w$$

(Answer 2)

$$(\lambda x.\ \lambda y.\ x)\ ((\lambda z.\ z)\ w) \xrightarrow[\beta]{} \lambda y.\ ((\lambda z.\ z)\ w) \xrightarrow[\beta]{} \lambda y.\ w$$

Problem 3 Write the output to the display when executing the following program in C++.

```
#include <stdio.h>
class B {
public:
  virtual char f() { return 'B';}
  char g() { return 'B'; }
```

```
  char testF(B *b) { return b->f();}
  char testG(B *b) { return b->g();}
};
class D : public B {
public:
  char f() { return 'D';}
  char g() { return 'D';}
};
int main(void) {
  D *d = new D;
  printf("%c%c\n", d->testF(d), d->testG(d));
  return 0;
}
```

(Answer)

```
DB
```

Problem 4 Write the solution (the substitution to the variable `X`) to the query `a(X).` after defining `a`, `b`, `c`, `d`, and `e` in Prolog as follows.

```
a(1) :- b.
a(2) :- e.
b :- !, c.
b :- d.
c :- fail.
d.
e.
```

(Answer)

```
X = 2
```

Problem 5

Show the meaning of the following programs (1) and (2) by using the rules presented in the lecture. Note that the programs are in the small subset of C presented in the lecture. Let the states before executing the programs both to be $\sigma = \{(\mathtt{X}, 3), (\mathtt{Y}, 1), (\mathtt{Z}, 0)\}$.

(1) `Z=(X+4);`

$$\frac{\dfrac{< \mathtt{X}, \sigma > \rightarrow 3 \quad < 4, \sigma > \rightarrow 4}{< (\mathtt{X} + 4), \sigma > \rightarrow 7}}{< \mathtt{Z} = (\mathtt{X} + 4);, \sigma > \rightarrow \sigma[7/Z]}$$

So in the state σ, after executing the program `Z=(X+4);` the state becomes as follows.

$$\sigma[7/\mathtt{Z}] = \{(\mathtt{X}, 3), (\mathtt{Y}, 1), (\mathtt{Z}, 7)\}$$

(2) `while(Y){Y=(Y-1);}`

$$\dfrac{< \mathtt{Y}, \sigma > \to 1 \quad \dfrac{\dfrac{< \mathtt{Y}, \sigma > \to 1 \quad < 1, \sigma > \to 1}{< (\mathtt{Y} - 1), \sigma > \to 0}}{< \mathtt{Y} = (\mathtt{Y} - 1);, \sigma > \to \sigma[0/\mathtt{Y}]} \quad \dfrac{< \mathtt{Y}, \sigma[0/\mathtt{Y}] > \to 0}{< \mathtt{while}(\mathtt{Y})\{\mathtt{Y} = (\mathtt{Y} - 1);\}, \sigma[0/\mathtt{Y}] > \to \sigma[0/\mathtt{Y}]}}{< \mathtt{while}(\mathtt{Y})\{\mathtt{Y} = (\mathtt{Y} - 1);\}, \sigma > \to \sigma[0/\mathtt{Y}]}$$

So in the state σ, after executing the program `while(Y){Y=(Y-1);}` the state becomes as follows.

$$\sigma[0/\mathtt{Y}] = \{(\mathtt{X}, 3), (\mathtt{Y}, 0), (\mathtt{Z}, 0)\}$$