Technische Universität München *Summer 2011*
Fakultät für Informatik *Kalmer Apinis*
Prof. Dr. H. Seidl *apinis@in.tum.de*

Virtual Machines

Exercise Sheet 3

Deadline: 31. May 2011, 14:00

Exercise 1: *Structs* *6 Points*

Give the address environment ρ for the following declarations. (For use in the next part of the exercise, also give an concrete address to `a`)

```
struct int_list {
  int              val;
  struct int_list *next;
};
struct A {
  int              buffer[64];
  struct int_list  x;
  int              y;
} a;
```

According to the scheme, translate (and inline ρ) the following expressions:

- `a.x.val + a.y`
- `a.buffer[a.x->next.val}`

Exercise 2: *Overlapping record names* *2 Points*

In the lecture we assumed, that all field names are unique. Now assume, that fields are unique inside a `struct` definition. What must be changed?

Exercise 3: *Function call* *9 Points*

Translate the following function to CMa (with an empty ρ)

```
int f(int x, int y){
  return x*y+2;
}
```

Translate the expression $e \equiv$ `f(8,5)` into CMa using the scheme from the lecture. Execute the statement e and draw the stack states before and after `mark`, `call`, `enter`, `alloc`, `return`, and `slide`.

Exercise 4: *Function call instructions* *3 Points*

Which of the function call instructions (`mark`, `call`, `enter`, `alloc`, `return`, and `slide`) can be implemented as macro instructions (— from previously defined instructions). Why? Give the macro definitions for instruction where it is possible.