1. Show how the lambda calculus expression $(\lambda x.\ \lambda y.\ (xy))(\lambda u.\ \lambda v.\ (uv))$ reduces to $\lambda y.\ \lambda v.\ (yv)$.
2. The following function `f` is said to be impure, show example invocations including their inputs, results and side-effects.

```c
void f(int *x, int y){
  *x = *x + y;
  printf("%d\n",*x);
  return;
}
int main(void) {
  int x = 0;
  f(&x, 3);
  x = x - 2;
  f(&x, 3);
}
```

3. Show the curried lambda form of the following multi-parameter function: `plus x y z = x + y + z`. *Bonus points if you can show the corresponding lambda calculus expression given some lambda expression* add *as an alias of the addition expression*.