13 Types (nk480)

(*a*) Using the simply-typed lambda calculus with the letcont primitive, give well-typed terms corresponding to proofs of the following classical tautologies:

(*i*) $dne : \neg\neg A \to A$ [3 marks]

(*ii*) $contra : (A \to B) \to (\neg B \to \neg A)$ [4 marks]

(*iii*) $demorgan_1 : \neg(A \vee B) \to \neg A \wedge \neg B$ [5 marks]

(*iv*) $demorgan_2 : (\neg A \wedge \neg B) \to \neg(A \vee B)$ [5 marks]

(*b*) (*i*) Briefly explain what the following Agda type says.

$$\forall \{\mathtt{A} : \mathtt{Set}\}\{\mathtt{n} : \mathtt{Nat}\} \to \mathtt{Vec}\,\mathtt{A}\,\mathtt{n} \to (\mathtt{i} : \mathtt{Nat}) \to (\mathtt{i} < \mathtt{n}) \to \mathtt{A}$$

[1 mark]

(*ii*) Given the two following Agda declarations:

$$\mathtt{zip} : \forall \{\mathtt{A} : \mathtt{Set}\}\{\mathtt{B} : \mathtt{Set}\}\{\mathtt{n} : \mathtt{Nat}\} \to (\mathtt{Vec}\,\mathtt{A}\,\mathtt{n} \times \mathtt{Vec}\,\mathtt{B}\,\mathtt{n}) \to \mathtt{Vec}\,(\mathtt{A} \times \mathtt{B})\,\mathtt{n}$$

$$\mathtt{unzip} : \forall \{\mathtt{A} : \mathtt{Set}\}\{\mathtt{B} : \mathtt{Set}\}\{\mathtt{n} : \mathtt{Nat}\} \to \mathtt{Vec}\,(\mathtt{A} \times \mathtt{B})\,\mathtt{n} \to \mathtt{Vec}\,\mathtt{A}\,\mathtt{n} \times \mathtt{Vec}\,\mathtt{B}\,\mathtt{n}$$

Write a type expressing that `unzip` followed by `zip` is the identity. [2 marks]